



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY

A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

**ROZŠÍŘENÍ LABORATORNÍCH MODELŮ PARKOVIŠTĚ A
BEDNY O MOŽNOST PŘIPOJENÍ NA DECENTRÁLNÍ
PERIFERII**

EXTENDING THE PARKING AND BOXES LABORATORY MODELS FOR CONNECTIVITY TO
DECENTRALIZED PERIPHERY

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Vlastimil Mancl

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Miroslav Jirgl, Ph.D.

BRNO 2017



Bakalářská práce

bakalářský studijní obor **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

Student: Vlastimil Mancl

ID: 173695

Ročník: 3

Akademický rok: 2016/17

NÁZEV TÉMATU:

Rozšíření laboratorních modelů Parkoviště a Bedny o možnost připojení na decentralní periferii

POKYNY PRO VYPRACOVÁNÍ:

1. Seznamte se s laboratorními modely Parkoviště a Bedny a tyto modely popište.
2. Popište komunikační protokol Profibus/DP.
3. Realizujte připojení vstupů a výstupů modelů na některou z periferií SIEMENS řady ET200.
4. Připojte modely přes uvedenou periferii k PLC SIEMENS S7-300 a vytvořte ukázkové aplikace v dostupných jazycích (LAD, STL, GRAPH 7).
5. Diskutujte rozdíly v jednotlivých přístupech programování.
6. Vytvořte vizualizaci.
7. Demonstrujte funkčnost.

DOPORUČENÁ LITERATURA:

Martinásková M., L. Šmejkal. Řízení programovatelnými automaty, vydavatelství ČVUT, Praha, 2004.

Termín zadání: 6.2.2017

Termín odevzdání: 29.5.2017

Vedoucí práce: Ing. Miroslav Jirgl, Ph.D.

Konzultant:

doc. Ing. Václav Jirsík, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato práce se zabývá popisem a připojením laboratorních modelů dopravníku beden a parkoviště na decentralizovanou periferii od firmy SIEMENS pomocí komunikace PROFIBUS DP. Kvůli tomu byly vytvořeny programy pro řízení modelů přes PLC v jazycích LAD, STL a GRAPH. To bylo dosaženo pomocí programu STEP7, určeném pro programování PLC od firmy SIEMENS. Část práce se věnuje popisu vytvořených vizualizací modelů v programu WinCC. Modely jsou zde popsány a zhodnoceny z hlediska funkčnosti a použitelnosti pro výuku. Výsledkem této práce jsou ukázkové zadání programovacích úloh a jejich řešení, připojitelnost laboratorních modelů na decentralizovanou periferii a jejich řízení pomocí PLC a ovládání modelu pomocí vizualizace na PC.

KLÍČOVÁ SLOVA

SIEMENS, PLC, ET200B, ET200L-SC, SIMATIC S7 300, STEP7, LAD, STL, GRAPH, decentralizovaná periferie, PROFIBUS, vizualizace, WinCC

ABSTRACT

This bachelor's thesis is about the description and connection of box conveyor and parking lot laboratory models to decentralized periphery made by SIEMENS with the usage of PROFIBUS DP communication standard. In order to do so, programs for controlling models were made in three different programming languages: LAD, STL and GRAPH. The programs were developed in engineering software SIMATIC STEP 7, created especially for programming PLCs made by SIEMENS company. Another part of this thesis is about the visualisation of laboratory models, made in WinCC software. The models were rated in terms of functionality and usability for study purposes. The results of this thesis are programmes made for controlling the models with working visualisation and a possibility of connecting these models to decentralized peripheries.

KEYWORDS

SIEMENS, PLC, ET200B, ET200L-SC, SIMATIC S7 300, STEP7, LAD, STL, GRAPH, decentralized periphery, PROFIBUS, visualization, WinCC

MANCL, Vlastimil *Rozšíření laboratorních modelů Parkoviště a Bedny o možnost připojení na decentrální periferii*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2017. 48 s. Vedoucí práce byl Ing. Miroslav Jirgl, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Rozšíření laboratorních modelů Parkoviště a Bedny o možnost připojení na decentrální periferii“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora(-ky)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Miroslavu Jirglovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

podpis autora(-ky)

OBSAH

| | |
|--|-----------|
| Úvod | 11 |
| 1 PROFIBUS DP | 12 |
| 1.1 Varianty Profibusu: | 13 |
| 1.2 Topologie Profibusu | 14 |
| 2 Popis použitých decentralizovaných periférií. | 15 |
| 2.1 Simatic ET 200L-SC | 15 |
| 2.1.1 Blok elektroniky | 15 |
| 2.1.2 Terminál | 16 |
| 2.2 Simatic ET 200B | 16 |
| 2.2.1 Blok elektroniky | 17 |
| 2.2.2 Terminál | 17 |
| 3 Řešení Úloh | 19 |
| 3.1 Zapojení úloh | 19 |
| 3.2 Úloha č.1 - Bedny | 20 |
| 3.2.1 Popis modelu | 20 |
| 3.2.2 HW konfigurace | 21 |
| 3.2.3 Zapojení výstupního konektoru modelu | 22 |
| 3.2.4 Vizualizace | 23 |
| 3.2.5 Vzorová úloha a řešení | 25 |
| 3.2.6 Závěr úlohy | 27 |
| 3.3 Úloha č.2 - Parkoviště | 28 |
| 3.3.1 Popis modelu | 28 |
| 3.3.2 HW konfigurace | 29 |
| 3.3.3 Zapojení výstupního konektoru modelu | 30 |
| 3.3.4 Vizualizace | 31 |
| 3.3.5 Vzorová úloha a řešení | 33 |
| 3.3.6 Závěr úlohy | 35 |
| 3.4 Porovnání programovacích jazyků | 36 |
| 3.5 Demonstrace funkčnosti vizualizace | 37 |
| 3.5.1 Model dopravníku beden | 37 |
| 3.5.2 Model parkoviště | 39 |
| 3.6 Úpravy modelu dopravníku beden | 42 |
| 4 Závěr | 45 |

| | |
|----------------------------------|----|
| Literatura | 46 |
| Seznam symbolů, veličin a zkratk | 47 |
| A Obsah přiloženého CD | 48 |

SEZNAM OBRÁZKŮ

| | | |
|------|--|----|
| 1.1 | ISO/OSI model komunikace všech variant PROFIBUSU [1] | 12 |
| 1.2 | Ukázka připojených stanic [3] | 13 |
| 1.3 | Princip přístupu k síti u sítě Profibus. [1] | 14 |
| 2.1 | Blok elektroniky ET 200L-SC 16 DI/16 DO DC 24 V/0.5 A [3] | 15 |
| 2.2 | Terminál s vývody se zasunutým modulem pro 32 digitálních vstupů. [3] | 16 |
| 2.3 | Blok elektroniky ET 200B-8DI/8DO [4] | 17 |
| 2.4 | Terminál TB1/DC 6ES7 [4] | 18 |
| 2.5 | Terminál - očíslování svorkovnic TB1/DC 6ES7. Pohled zepředu. [4] . | 18 |
| 3.1 | Schéma zapojení úloh | 19 |
| 3.2 | Model původního dopravníku beden | 20 |
| 3.3 | Univerzální řídicí deska | 21 |
| 3.4 | Konfigurace PLC - lišta | 21 |
| 3.5 | Konfigurace PLC - výrobní čísla, adresy | 22 |
| 3.6 | Konfigurace DP - adresy | 22 |
| 3.7 | Vývod z modelu - konektor CANON 25, zásuvka pohled shora | 22 |
| 3.8 | Rozhraní vizualizace pro model dopravníku beden | 23 |
| 3.9 | Kód pro generování 100ms pulzů | 24 |
| 3.10 | Stavový diagram pro řízení modelu dopravníku beden | 26 |
| 3.11 | Model parkoviště | 28 |
| 3.12 | Konfigurace PLC - lišta | 29 |
| 3.13 | Konfigurace PLC - výrobní čísla, adresy | 29 |
| 3.14 | Konfigurace DP - adresy | 30 |
| 3.15 | Vývod z modelu - konektor CANON 25, zásuvka pohled shora | 30 |
| 3.16 | Rozhraní vizualizace pro model parkoviště | 32 |
| 3.17 | Stavový diagram pro řízení modelu parkoviště | 34 |
| 3.18 | Kompenzace a řízení motoru závor. | 35 |
| 3.19 | Kontrolní panel z pravého horního rohu rozhraní. | 37 |
| 3.20 | Ukázka sepnutí solenoidu a senzoru. | 37 |
| 3.21 | Kontrolní panel z pravého horního rohu rozhraní. | 38 |
| 3.22 | Kontrolní panel z pravého horního rohu rozhraní. | 39 |
| 3.23 | Reprezentace vozidel před bránou. | 39 |
| 3.24 | Vyobrazení stavu závor. | 40 |
| 3.25 | Průjezd vozidla. | 40 |
| 3.26 | Textové hlášky o stavu parkoviště, brány a vozidla | 41 |
| 3.27 | Zapojení stabilizátoru pevného napětí STMicroelectronics 7812CV-DG | 42 |

| | | |
|------|--|----|
| 3.28 | Blokový diagram zapojení univerzální desky u modelu dopravníku beden. | 44 |
| 3.29 | Výsledná podoba modelu dopravníku beden. | 44 |

SEZNAM TABULEK

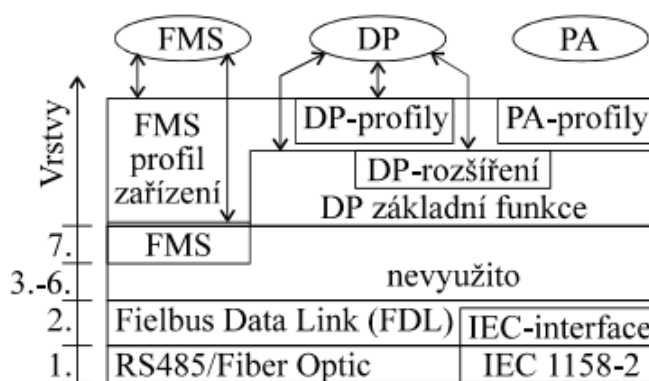
| | | |
|-----|--|----|
| 3.1 | Tabulka vstupů a výstupů a připojení na svorkovnice ET | 25 |
| 3.2 | Tabulka vstupů a výstupů a připojení na svorkovnice ET | 31 |

ÚVOD

Tato práce se věnuje problematice průmyslové komunikace Profibus DP (Decentralizovaná Periferie nebo také Distributed Periphery) a její aplikace v laboratorních úlohách Bedny a Parkoviště, sloužících pro výuku. Konkrétně se zde používají přístroje od firmy Siemens a to ET 200B a ET 200L-SC. Jejich popisu bude věnována část semestrální práce. Bude zde také popis laboratorních modelů a vytvoření ukázkových zadání, které budou naprogramovány v jazycích dostupných pro PLC od firmy Siemens tj. LAD, STL a GRAPH. Jednotlivé jazyky pak zde budou porovnány z pohledu přístupu k programování a jejich vhodnosti pro danou úlohu. Pro oba modely bude následně vytvořena a popsána vizualizace v programu WinCC flexible. Výpisy z programů přidám k práci v elektronické podobě jako přílohu na CD.

1 PROFIBUS DP

PROFIBUS (Process field bus) byl vyvinut v Německu několika významnými firmami pod koordinací DBFT (Deutsches Bundesministerium für Forschungs und Technik). V Německu se taktéž stal normou DIN 19245. Profibus v ISO/OSI modelu implementuje pouze 3 vrstvy a to 1,2 a 7. Jako Fyzické médium se používá stíněná kroucená dvoulinka s rozhraním RS 485 s max. délkou 1200m (4800m s opakovací) nebo optické vlákno. Pro variantu PA (Process Automation) pak proudová smyčka. Přenosová rychlost skýtá širokou škálu rychlostí 1,5 Mb/s, 3 Mb/s, 6 Mb/s, 9 Mb/s a 12 Mb/s a více (Profibus DP). Profibus PA má pouze jedinou rychlost dle normy IEC 1158-2 a to 31,25 kb/s. [2]



Obr. 1.1: ISO/OSI model komunikace všech variant PROFIBUSU [1]

Popis vrstev modelu:

- 1. Fyzická vrstva:** tato vrstva zaobaluje fyzické spojení, mechanické, elektrické vlastnosti, topologie sítě a rozhraní. Jako fyzické médium používá Profibus rozhraní RS 485 s nestandardně definovanými parametry nebo taktéž může využívat přenos světlovodičem. Profibus PA je pak upraven pro výbušné zóny tak, že používá proudovou smyčku (standard IEC 1158-2.).
- 2. Linková vrstva:** tato vrstva zaopatrňuje přístup k síti a zabezpečuje tvorbu zprávy, tvořené tokem bitů (11-ti bitové kódování znaků typu UART skýtá: jeden start bit, jeden stop bit, 8 informačních bitů a jeden paritní bit (sudá parita)). Profibus využívá Token bus s přístupem Master Slave.
- 7. Aplikační vrstva:** tato vrstva zaopatrňuje nezbytné služby pro komunikaci. Má dvě podvrstvy, FMS (Fieldbus Message Specification) a LLI (Lower Layer Interface). [2]

1.1 Varianty Profibusu:

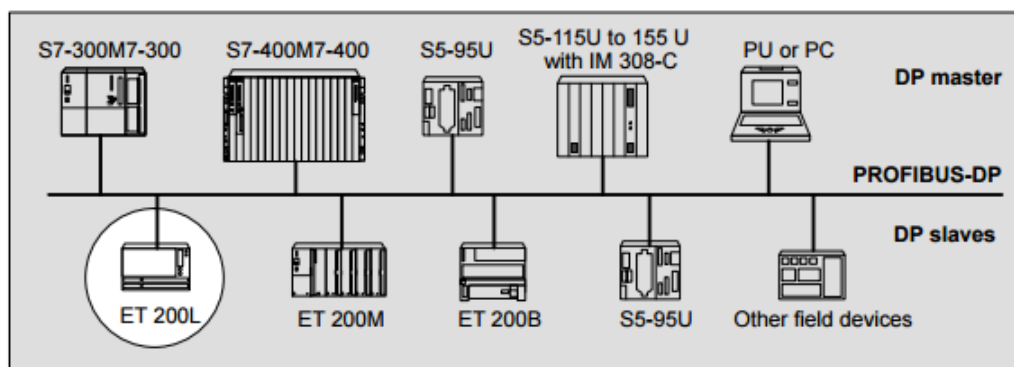
Byly vyvinuty tři varianty Profibusu. Profibus DP, Profibus FMS a Profibus PA.

Profibus DP - tahle verze je nejjednodušší verzí a je taktéž nejpoužívanější. Profibus DP je určený pro komunikaci typu Master-Slave. Používá se převážně pro jeho rychlý přenos signálů z decentralizovaných periférií a vzdálených I/O periférií. Fyzické médium využívá RS 485 nebo světlovodič s přenosovou rychlostí až 12Mb/s. Přenos zaopatrňuje jednoduchý soubor bloků a funkcí. V mé semestrální práci používám právě tuto komunikaci.

Profibus FMS - díky široké nabídce služeb pro práci s daty, programy a alarmy, se FMS používá pro komunikaci v heterogenním prostředí na vyšší úrovni. Fyzické médium pro přenos je shodná s verzí DP. Rychlost má ale kvůli většímu počtu služeb nižší.

Profibus PA - díky tomu, že se jako fyzické médium pro přenos používá proudová smyčka, je verze PA vhodná pro řízení procesů i v prostředí s nebezpečím výbuchu. [2]

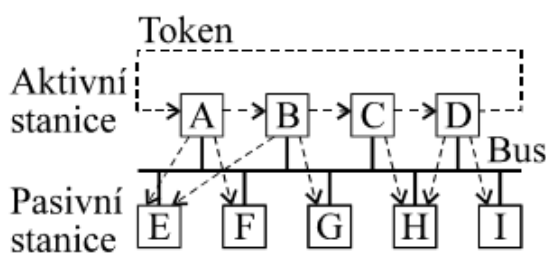
Standard RS-485 dovoluje připojit maximálně 32 stanic. U Profibusu se používají tzv. opakovače, které tohle omezení eliminují. Opakovači můžeme propojit až 10 segmentů, při celkovém počtu 127 stanic. Každý segment je třeba zakončit aktivním terminátorem.



Obr. 1.2: Ukázka připojených stanic [3]

1.2 Topologie Profibusu

V topologii Profibusu se dělí připojené stanice na pasivní a aktivní. Aktivní stanice (PLC, PC apod) si mezi sebou předávají povolení vysílat (token). Mezi pasivními stanicemi k žádnému předávání tokenu nedochází. Pasivní stanice komunikují jen v době, když jsou vyzvány od některé z aktivních stanic s tokenem. Z obrázku 1.3 je patrné zapojení. Stanice A, B, C, D jsou aktivní stanice. Zbylé stanice na obrázku jsou pasivní. Pasivní stanice čekají na výzvu od aktivní stanice. Na obrázku 1.2 je pak ukázka toho, co se například může připojit k síti v rámci přístrojů od firmy SIEMENS.



Obr. 1.3: Princip přístupu k síti u sítě Profibus. [1]

Aktivní stanice vědí od koho jim připutoval token a komu ho mají předat. Doba, po kterou aktivní stanice vlastní token, je parametrizovatelná. [2]

2 POPIS POUŽITÝCH DECENTRALIZOVANÝCH PERIFERIÍ.

Pro řízení laboratorních modelů jsem vybral dva přístroje. Simatic ET 200B pro řízení modelu dopravníku beden a Simatic ET 200L-SC pro řízení modelu parkoviště.

2.1 Simatic ET 200L-SC

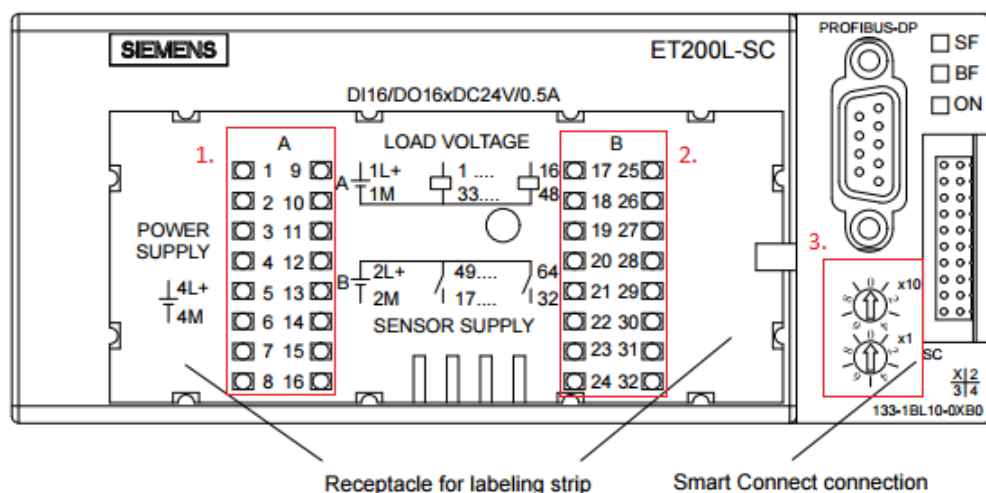
Mnou použitá periférie se skládá ze dvou částí, z bloku elektroniky a z terminálu s vývody, do kterého je blok zasunut. Blok elektroniky určuje počet vstupů a výstupů, které budou k dispozici a taktéž obsluhuje komunikaci přes PROFIBUS DP.

2.1.1 Blok elektroniky

Úplný anglický název: Electronics Block L-SC 16 DI/16 DO DC 24 V/0.5 A
Výrobní číslo: 6ES7 133-1BL10-0XB0

Blok obsahuje:

- 16 digitálních vstupů - DC 24 V
- 16 digitálních výstupů - 0,5 A a DC 24 V
- TB 16SC konektor



Obr. 2.1: Blok elektroniky ET 200L-SC 16 DI/16 DO DC 24 V/0.5 A [3]

Na obrázku 2.1 je zobrazen blok elektroniky. V části 1 se nacházejí kontrolní LED diody pro výstupy, v části 2 pak kontrolní LED diody pro vstupy. V části

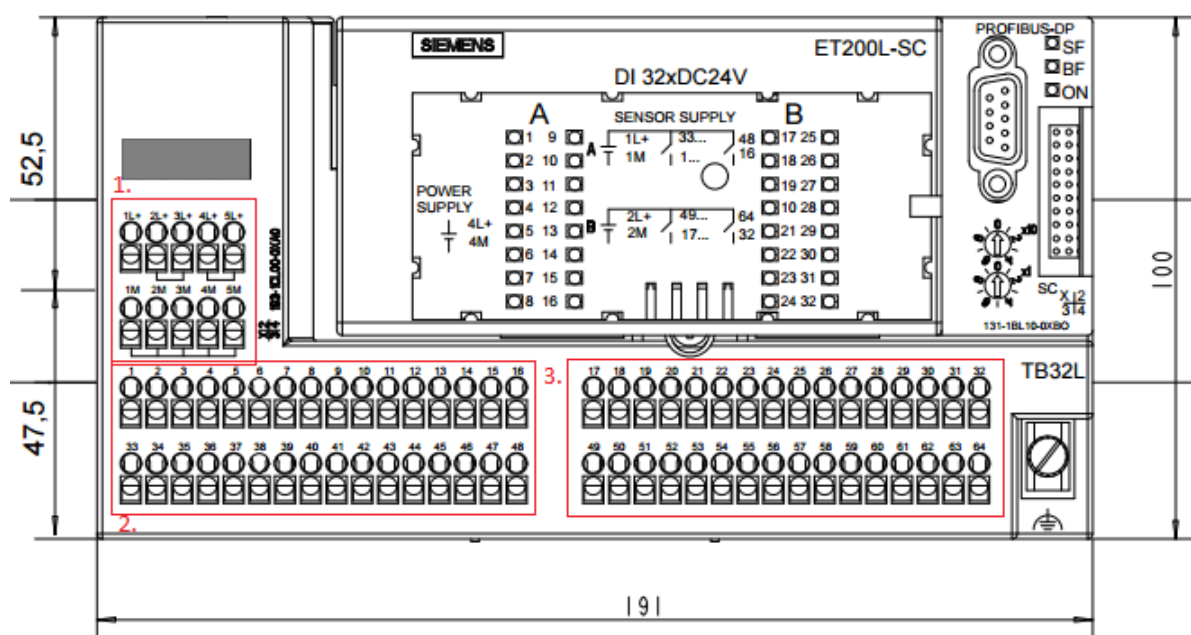
3 se otočným přepínačem nastavuje adresa zařízení. Dále se zde nachází schéma s očíslovaným zapojením vstupů a výstupů, stavové informační LED diody, konektor pro profibus a konektor pro SC (Smart Connect), který lze použít na rozšíření počtu vstupů a výstupů pomocí terminálu TV 16SC, do kterého lze zasunout až 8 SC modulů (mohou být analogové i digitální). [3]

2.1.2 Terminál

Úplný anglický název: Terminal Block TB 32L

Výrobní číslo: 6ES7 193-1CL00-0XA0

Na obrázku 2.2 je zobrazen terminál. V části 1. se nacházejí svorky pro napájení. Vrchní řada kladné svorky, spodní řada záporné (zemní) svorky. V části 2. jsou výstupní svorky, v části 3. pak svorky vstupní.



Obr. 2.2: Terminál s vývody se zasunutým modulem pro 32 digitálních vstupů. [3]

2.2 Simatic ET 200B

Námi použitá druhá periférie se rovněž skládá ze dvou částí. Z bloku elektroniky a z terminálu. Platí pro ni obdobný popis jako u předchozí periférie. Liší se tvarem, množstvím vstupů/výstupů a provedením.

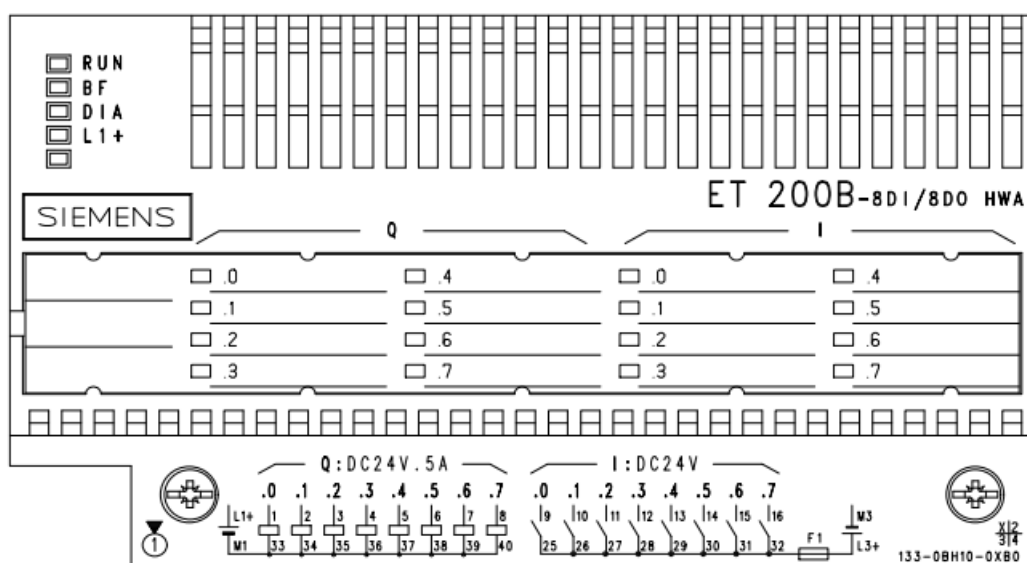
2.2.1 Blok elektroniky

Úplný anglický název: Electronics Block ET 200B-8DI/8DO

Výrobní číslo: 6ES7 133-0BH01-0XB0

Blok obsahuje:

- 8 digitálních vstupů - DC 24 V
- 8 digitálních výstupů - DC 24 V
- Výstupy lze v programu deaktivovat.
- spoždění vstupu: 3 ms



Obr. 2.3: Blok elektroniky ET 200B-8DI/8DO [4]

Na obrázku 2.3 je zobrazen blok elektroniky. V levém horním rohu jsou stavové informační LED diody. Ve středu je informační panel s LED diodami pro vstupy a výstupy. Ve spodní části je umístěno principiální schéma s očíslovaným zapojením vstupů a výstupů. [4]

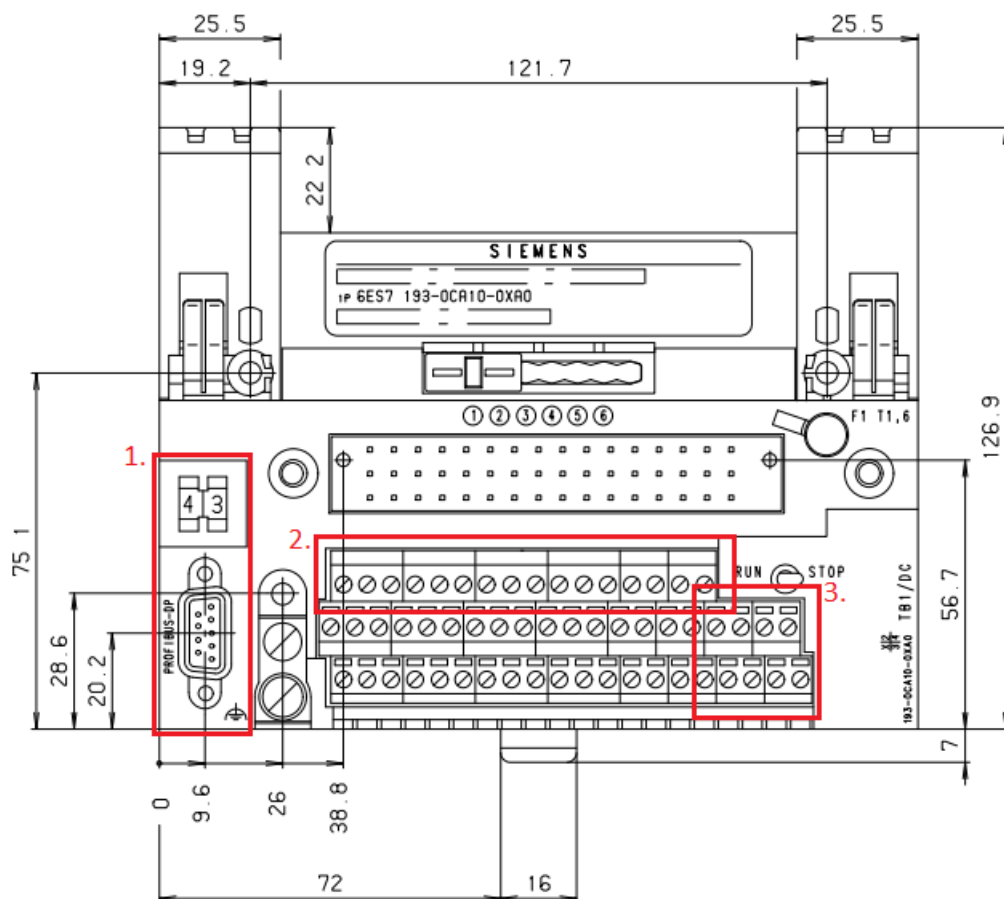
2.2.2 Terminál

Úplný anglický název: Terminal Block TB1/DC

Výrobní číslo: 6ES7 193-0CA10-0XA0

Narozdíl od terminálu TB32L obsahuje terminál TB1/DC mimo vývodů i číselník pro nastavení adresy DP a konektor pro PROFIBUS, jak lze vidět na obrázku 2.4 v části 1. V 2. části se nachází svorkovnice pro vstupy a výstupy. Pro námi použitý

blok elektroniky platí, že prvních 8 (zleva) je výstupních. Následujících osm pak vstupních. V části tři můžeme najít svorky pro napájení. Vrchní čtyři svorky jsou pro kladné napětí spodní 4 pak pro záporné (zem).



Obr. 2.4: Terminál TB1/DC 6ES7 [4]



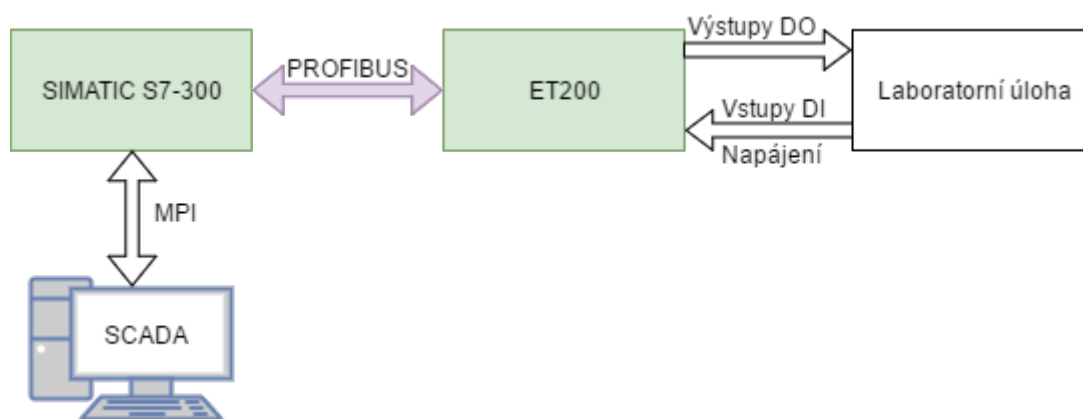
Obr. 2.5: Terminál - očíslování svorkovnic TB1/DC 6ES7. Pohled zepředu. [4]

3 ŘEŠENÍ ÚLOH

Dle zadání jsem řešil úlohy pro modely Parkoviště a Bedny. Původně jsou modely řešeny tak, že jejich vstupy a výstupy jsou připojené přímo na vstupní a výstupní karty PLC SIMATIC S7 300. Musel jsem tedy projít dokumentace modelů pro zjištění zapojení vývodního konektoru a vytvořit protikus v podobě konektoru a plochého kabelu, jehož vývody jsem připojil na vstupy/výstupy decentralizované periférie. V případě modelu parkoviště nebyla jeho dokumentace dobře čitelná a pochopitelná, proto se výstupní konektor musel proměřit, aby se přesně určily piny vstupů a výstupů jednotlivých komponent na modelu.

3.1 Zapojení úloh

Pro řízení obou modelů jsem použil PLC od firmy SIEMENS S7 300. Hw konfigurace je uvedená u každé úlohy zvlášť. Pro připojení vstupů a výstupů byly použity výše zmíněné přístroje ET 200L-SC a ET 200B. Konkrétně jsem pro model beden zvolil ET 200B z důvodu, že v téhle úloze mi stačil počet vstupů a výstupů na periférii. Jelikož model parkoviště měl víc než 8 vstupů, bylo potřeba zvolit ET 200L-SC. Principiální schéma je na obrázku 3.1. Nutností pro splnění zadání byla tvorba vlastních protikusů k vývodům z modelů. Použil jsem vidlici CANON 25, ze které jsem vyvedl dvacetí žilový plochý kabel. Vývody pak byly upevněny do svorek ET. Periférie byly napájeny přímo z napájení modelů. K propojení PLC a ET byl použit standartní kabel pro PROFIBUS DP. K PLC pak bylo připojeno PC přes rozhraní MPI. Na PC byla přes program WinCC flexible spuštěná vizualizace.

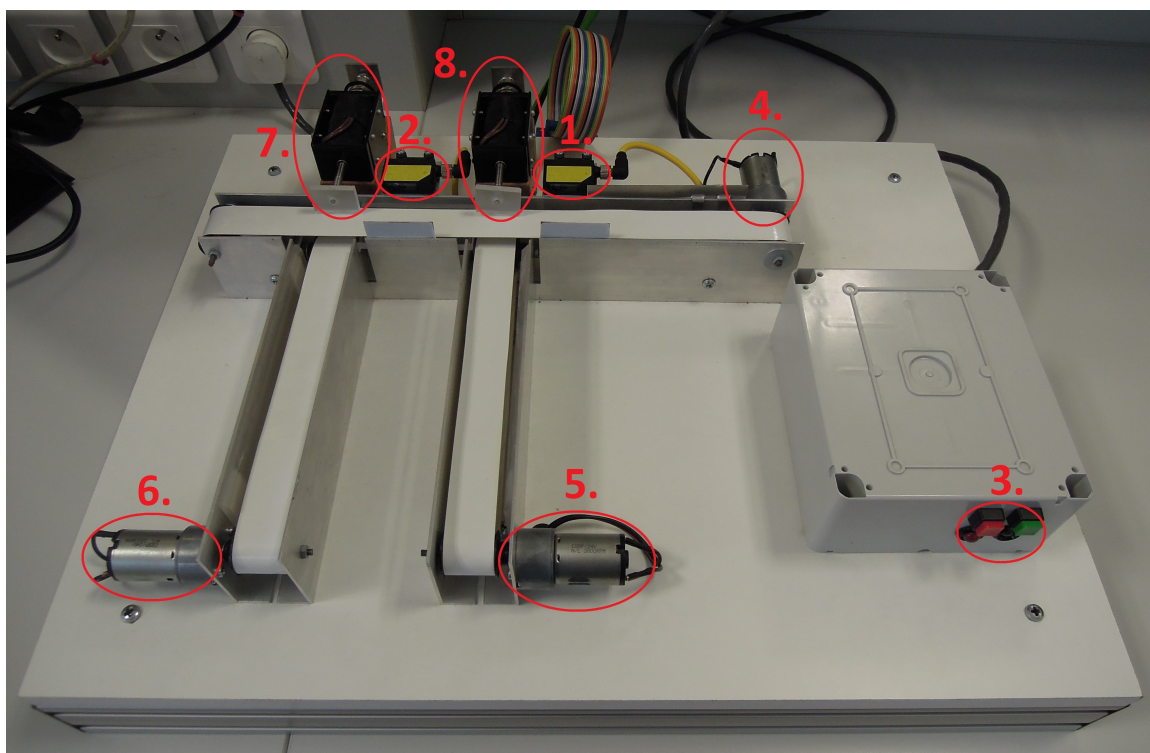


Obr. 3.1: Schéma zapojení úloh

3.2 Úloha č.1 - Bedny

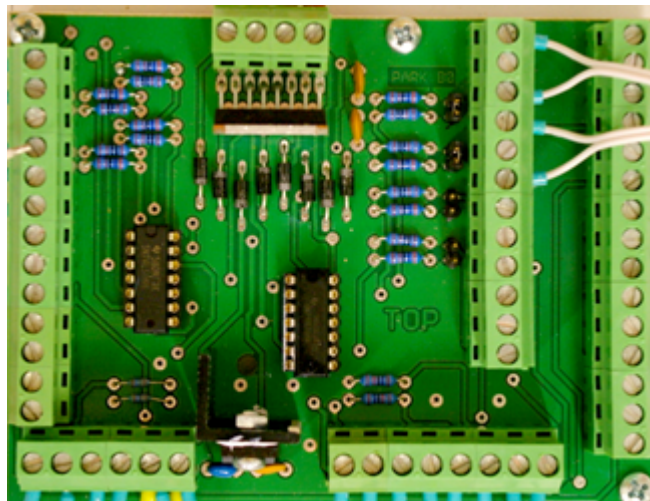
3.2.1 Popis modelu

Model je zachycen na obr. 3.2. Je tvořen třemi dopravními pásy, poháněnými třemi motory. U nejdelšího (hlavního) pásu se nachází dva solenoidy sloužící k odsunování beden na příslušný vedlejší pás. Napravo od solenoidů jsou umístěné optické závory, sloužící k detekci beden. Dále pak můžeme na modelu najít panel s hlavním vypínačem, dvěma tlačítky a dvěma LED diodami. Vývod modelu je tvořen zásuvkou CANON 25 vyvedeným na krátkém plochém kabelu. Popis pozic označených na obrázku 3.2 je v tabulce 3.1



Obr. 3.2: Model původního dopravníku beden

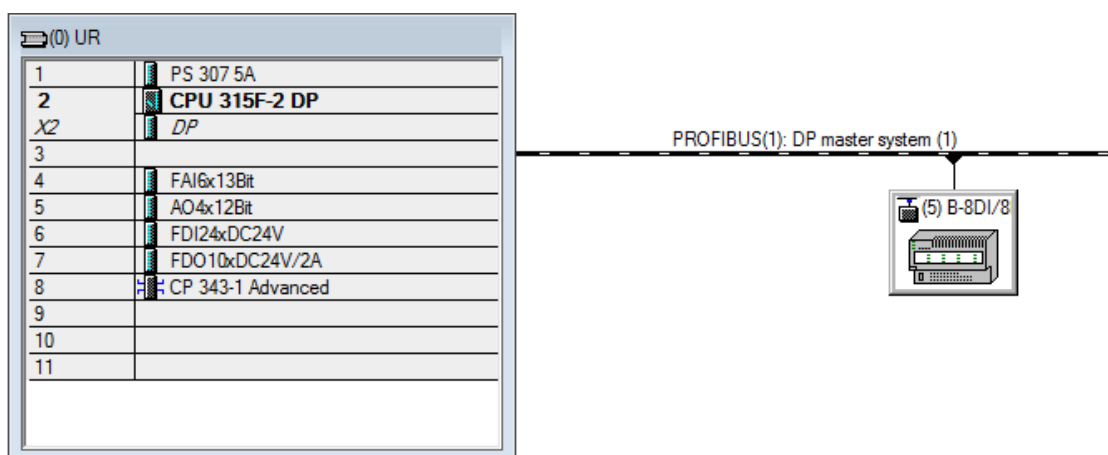
Motory jsou ovládány pomocí univerzální řídicí desky viz. obrázek 3.3, která zpracovává signály z decentralizované periferie. Na desku jsou připojeny i informační LED diody a tlačítka. Optické senzory jsou napájeny z desky, která taktéž předává výstupní signál senzorů do PLC. Solenoidy jsou ovládány zvlášť pomocí vlastních spínačů. Blokové schéma je znázorněno na obrázku 3.28. Vstupy/výstupy pro připojení k PLC jsou pomocí krátkého plochého kabelu vyvedeny od univerzální desky na konektor CANON 25. Detailní schéma řídicí desky viz. elektronická příloha na CD. Celá úloha je napájena vlastním zdrojem připojeným k síti 230V/50Hz s výstupem 24V ss.



Obr. 3.3: Univerzální řídicí deska

3.2.2 HW konfigurace

Řídicí jednotku (Master) jsem si zvolil PLC SIMATIC S7 300. Jako slave pro připojení vstupů a výstupů z modelu byla zvolena decentralizovaná periferie ET 200B. Výsledná konfigurace byla následující:



Obr. 3.4: Konfigurace PLC - lišta

Na obrázku 3.4 je vidět, že na první pozici na liště byl zdroj PS 307 5A, na druhé pozici procesor 315F-2 DP. Třetí pozice je prázdná. Na čtvrté pozici je modul s analogovými vstupy FAI6x13bit, na páté pozici pak analogové výstupy AO4x12Bit. Následují moduly pro digitální vstupy a výstupy FDI24xDC24V a FDO10xDC24V/2A. Na poslední pozici je síťový modul CP 343-1 Advanced.

Z přídatných modulů pro analogové či digitální vstupy/výstupy jsem žádný nevyužil. Použil jsem digitální vstupy a výstupy decentralizované periferie (DP). Mnou

| Slot | Module | Order number | Firmware | MPI address | I address | Q address | Comment |
|------|----------------------|----------------------------|-------------|-------------|-----------|-----------|---------|
| 1 | PS 307 5A | 6ES7 307-1EA00-0AA0 | | | | | |
| 2 | CPU 315F-2 DP | 6ES7 315-6FF01-0AB0 | V2.0 | 2 | | | |
| 3 | | | | | 204,7" | | |
| 4 | AI6x13Bit | 6ES7 336-1HE00-0AB0 | | | 8...23 | 8...11 | |
| 5 | AO4x12Bit | 6ES7 332-5HD01-0AB0 | | | | 272...279 | |
| 6 | DI24xDC24V | 6ES7 326-1BK00-0AB0 | | | 24...33 | 24...27 | |
| 7 | DO10xDC24V/2A | 6ES7 326-2BF01-0AB0 | | | 40...45 | 40...47 | |
| 8 | CP 343-1 Advanced | 6GK7 343-1GX21-0XE0 | V1.0 | 3 | 448...463 | 448...463 | |
| 9 | | | | | | | |
| 10 | | | | | | | |
| 11 | | | | | | | |

Obr. 3.5: Konfigurace PLC - výrobní čísla, adresy

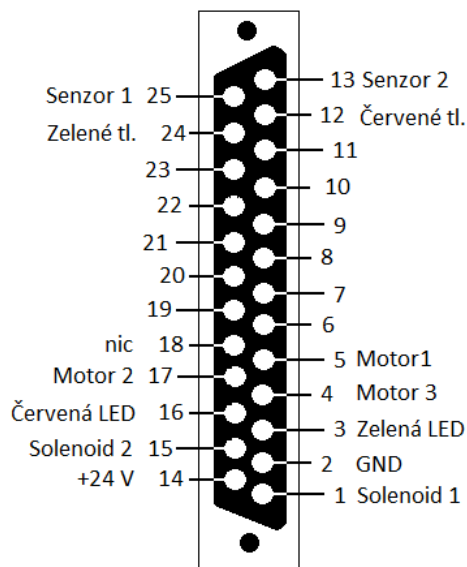
zvolený přístroj ET 200B má 8 DI a 8 DO, což pro tuhle úlohu stačí. Adresy vstupu začínají na nultém bytu, adresy výstupu na bytu prvním, viz. obr. 3.6.

| Slot | DP ID | ... | Order Number / Designation | I Address | Q Address | Comment |
|------|-------|-----|----------------------------|-----------|-----------|---------|
| 1 | 16DO | | 2 Byte Out, 1 Byte In | | 1...2 | |
| 2 | 8DI | | 2 Byte Out, 1 Byte In | 0 | | |

Obr. 3.6: Konfigurace DP - adresy

3.2.3 Zapojení výstupního konektoru modelu

Vývod je tvořen zásuvkou CANON 25. Nutností pro splnění zadání bylo vytvořit protikus, kterým jsem pak následně spojil model s ET 200B.

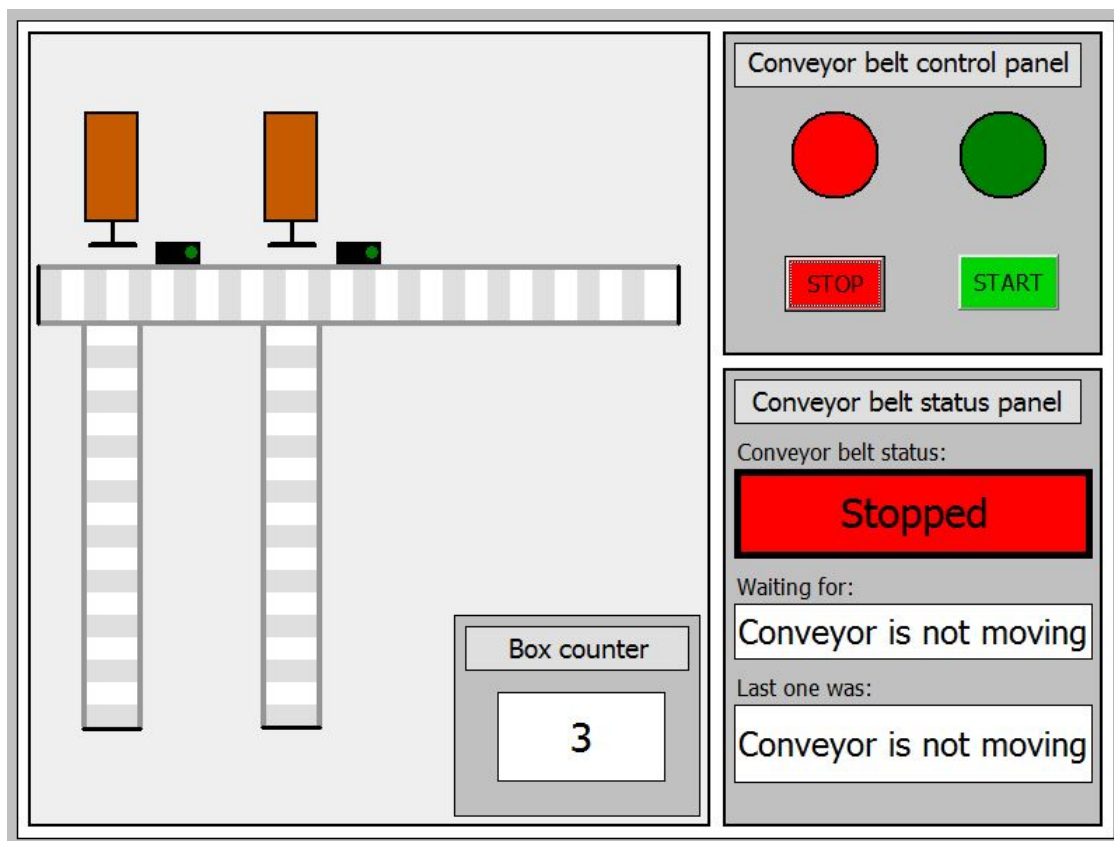


Obr. 3.7: Vývod z modelu - konektor CANON 25, zásuvka pohled shora

Propojení s DP je zaznačeno v tabulce 3.1. Jeden vodič plochého kabelu propojuje vždy příslušný pin s příslušnou svorkou, které pak odpovídá adresa a tag v programu uvedené taktéž v tabulce 3.1. ET je napájeno přímo z modelu.

3.2.4 Vizualizace

Vizualizace běžela v režimu runtime na PC a byla vytvořena v programu WinCC flexible. Rozhraní vizualizace je na obrázku 3.8.



Obr. 3.8: Rozhraní vizualizace pro model dopravníku beden

Popis rozhraní vizualizace

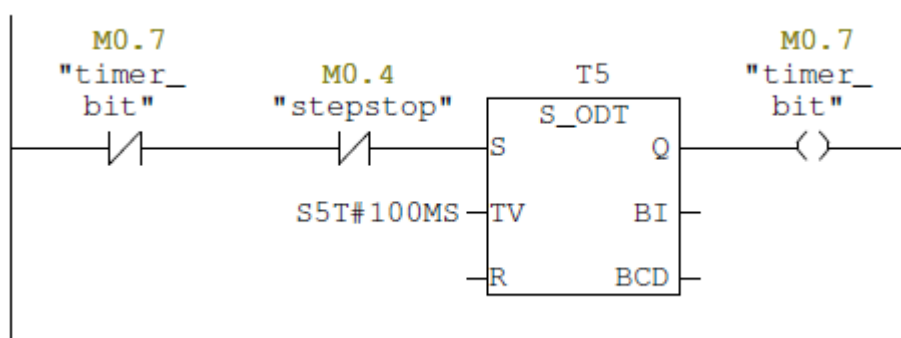
Rozhraní jsem vytvořil dle reálného modelu. V levé části je zachycen pohled shora na model dopravníku beden. Tři dopravní pásy mají vlastní animaci pohybu. K vytvoření animace jsem použil skupinu za sebou jdoucích šedých obdélníků, které se půlkou překrývají, umístěných na bílém pozadí. Kdy řada obdélníků je rozdělena do skupin po čtyřech a vždy se zobrazuje jen jeden z těchto čtyř obdélníků. Program, který obsluhuje veličinu na jejímž základě se zobrazuje příslušný obdélník, je popsán v další kapitole.

U hlavního pásu jsou modely dvou čidel, které změnou své barvy imitují funkci reálného čidla tj. indikace přítomnosti bedny. Nalevo od každého čidla je model solenoidu, který má vlastní animaci vysunutí ramena. Animace je dvoustavová, odrážející reálný stav solenoidu. Mezi vedlejším pásem a panely je čítač beden.

V pravém horním rohu rozhraní je ovládací panel s dvěma informačními diodami a dvěma tlačítky, který imituje reálný panel na modelu. Pod panelem jsem vytvořil informační panel, v němž jsou uvedené informace o stavu dopravního pásu. A pak informace o tom, na kterou bednu se čeká, a která bedna prošla jako poslední. Tahle část je zde hlavně z důvodu budoucího rozšíření. Zejména pak o konkrétní jména produktů, které se třídí. Pro mé zadání jsou bedny popsány jen jako první, druhá a třetí.

Popis funkčního bloku pro ovládání vizualizace

Ovládání proměnných realizuji ve funkčním bloku 2. Je zde generování 100 ms pulsů, viz obrázek 3.9, díky kterým se mění stav čítače od 0 do 4. Stav čítače je pak překonvertován a na základě tohoto překonvertovaného čísla se pak ve vizualizaci zobrazuje vždy jeden ze 4 obdélníků (viz předchozí kapitola). Pokud pásy nejedou, pulsy se ne-generují, proto nedochází k změně stavu čítače, čímž ani nedochází k pohybu pásu ve vizualizaci. Dále je zde naprogramován algoritmus pro vyhodnocování, která bedna přijela, a na kterou se čeká. K tomu jsem použil stavový automat, kde každému stavu odpovídá číslo proměnné "box_status" na jejímž základě se pak zobrazuje příslušný text z textového pole, vytvořeném ve vizualizaci. Byly použity dvě textové pole. Jedno pro poslední bednu a druhé pro očekávanou bednu. Funkčnost je zachycena v animaci v příloze na CD.



Obr. 3.9: Kód pro generování 100ms pulsů

3.2.5 Vzorová úloha a řešení

Zadání vzorové úlohy

Navrhněte řízení modelu dopravního pásu beden. Model se skládá ze třech pásů. Po páse poháněném motorem v části 4. (viz. 3.2) jsou posílány bedny. Bedny je nutno roztrždit a to tím způsobem, že:

- první bedna je odsunuta solenoidem na pás, poháněném motorem v části 5.
- druhá bedna je odsunuta solenoidem na pás, poháněném motorem v části 6.
- třetí bedna pokračuje na původním pásu.

Dále vytvořte počítadlo beden. Celé řízení navrhněte s použitím decentralizované periferie ET 200B.

| Název | Svorka ET | Adresa | Tag v programu | Pozice viz. obr. 3.2 |
|-------------|-----------|--------|----------------|----------------------|
| Sensor 1 | 9 | I0.0 | sensor_right | 1. |
| Sensor 2 | 10 | I0.1 | sensor_left | 2. |
| Červené tl. | 11 | I0.2 | btn_red | 3. |
| Zelené tl. | 12 | I0.3 | btn_green | 3. |
| Motor 1 | 1 | Q1.0 | mot1 | 4. |
| Motor 2 | 2 | Q1.1 | mot2 | 5. |
| Motor 3 | 3 | Q1.2 | mot3 | 6. |
| Červená LED | 4 | Q1.3 | light_red | 3. |
| Zelená LED | 5 | Q1.4 | light_green | 3. |
| Solenoid 2 | 6 | Q1.5 | sol_left | 7. |
| Solenoid 1 | 7 | Q1.6 | sol_right | 8. |

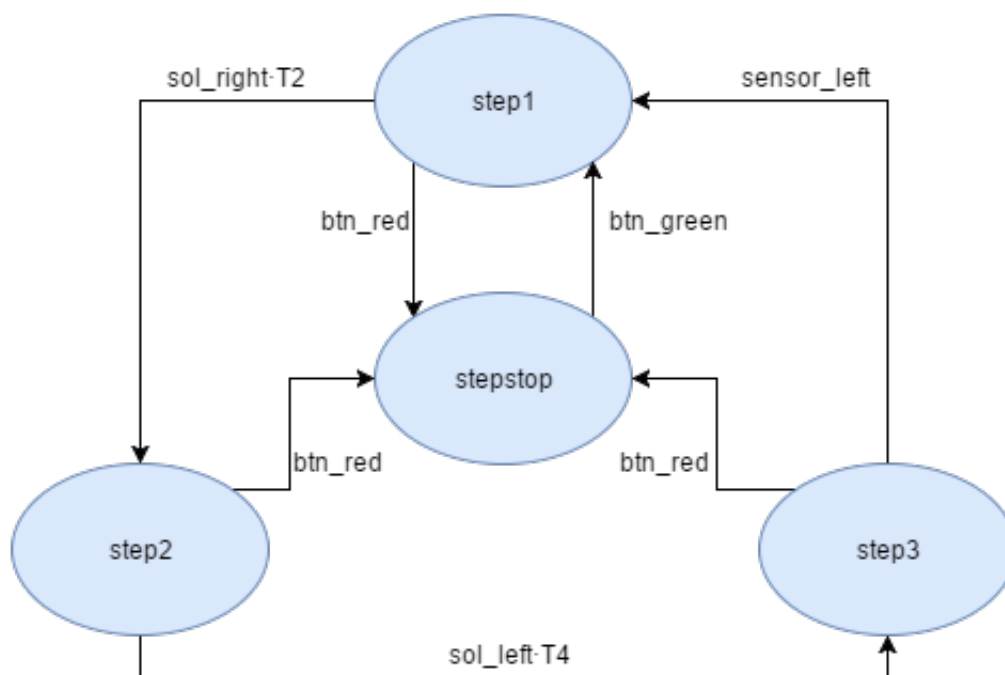
Tab. 3.1: Tabulka vstupů a výstupů a připojení na svorkovnice ET

Popis funkčních bloků v programu

Program obsahuje 4 funkční bloky a jeden organizační, ve kterém jsou potřebné funkční bloky volány. Ve FB1 implementuji ovládání modelu stavovým automatem v jazyce LAD. Ve FB2 je v jazyce LAD naprogramováno řízení proměnných potřebných pro ovládání vizualizace. Ve FB3 je v jazyku STL vytvořen program pro řízení modelu pomocí stavového automatu a ve FB4 je pak vytvořený program v jazyku GRAPH taktéž pro řízení modelu.

Popis stavového automatu v jazycích LAD a STL

Stavový automat je na obr. 3.10. Program byl vytvořen shodným přístupem jak v jazyku LAD, tak v jazyku STL a proto je shodný pro oba jazyky a taky proto si jsou



Obr. 3.10: Stavový diagram pro řízení modelu dopravníku beden

kódy podobné. Mé řešení má 4 stavy. Začíná se ve stavu stepstop. V tomto stavu jsou všechny motory zastavené a svítí červená LED dioda. Ze stavu step_stop se do stavu step1 dostaneme zmáčknutím zeleného tlačítka. Ve stavu step1 se rozeběhnou všechny motory a sleduje se signál z pravého sensoru. Jakmile je u něj bedna, tak se nastaví bit timer_enable, spouštějící časovač, po kterém se sepne pravý solenoid. Solenoid se po časovači T2 resetuje a tím se sepne stav step2, ve kterém se program opakuje s tím rozdílem, že se sleduje levý sensor. Pro přechod ze stavu 2 do stavu 3 se hlídá, zdali byl sensor zpět v úrovni 0, aby nedošlo k chybné detekci třetí bedny. Ze stavu step3 se pak dostaneme po příchodu bedny k levému senzoru do stavu step1. V každém bodě programu je možné se dostat do stavu step_stop zmáčknutím červeného tlačítka, díky čemuž zastavíme chod všech motorů. Opětovným zmáčknutím zeleného tlačítka se znovu dostáváme do stavu step1. Při přechodu stavů se inkrementuje stav čítače beden. Při přechodu ze stavů step1-3 do stavu step_stop je hodnota čítače zachována. Vynuluje se až při přechodu ze stavu step_stop do stavu step1. Kompletní výpis programů viz elektronická příloha na CD.

Popis stavového automatu v jazyku GRAPH

V jazyku GRAPH se stavový automat implementoval rozdílným přístupem, ale základní myšlenka je stejná a i zde platí diagram na obr. 3.10, ovšem jen obrazně, protože jednotlivé stavy byly rozděleny na více kroků. Celá posloupnost kroků pak

čítá 8 nerozvětvených kroků. V OB1 je při volání FB4 možnost přiřadit do INIT_SQ := proměnnou, která při změně úrovně z 0 na 1 zapříčiní, že se program vrátí do počátečního stavu. Díky tomuto jsem splnil vrácení automatu do stavu "step_stop" při stisku červeného tlačítka jak ve vizualizaci, tak na modelu. Krok 1 je ekvivalent stavu step_stop. Při zmáčknutí zeleného tlačítka se program dostane do kroku dva. V kroku dva se rozjedou pásy a rozsvítí se zelená informační led dioda, zhasne červená a čeká se, až pravý senzor zaznamená průchod krabice. Po průchodu se aktivuje krok s časovačem, po kterém se aktivuje krok 4, který aktivuje solenoid, inkrementuje stav čítače beden a aktivuje druhý časovač, aby měl solenoid dostatečný čas na reakci. Po uplynutí daného času se aktivuje pátý krok, ve kterém se čeká na indikaci levého senzoru. Poté se aktivuje šestý krok, ve kterém je čítač opět inkrementován a také se zde aktivuje další časovač. Zde se čeká, až bude levý senzor opět v úrovni 0, aby nedošlo k chybné detekci bedny. V dalším kroku se čeká na příchod bedny k levému senzoru. Při jejím příchodu se inkrementuje čítač a celá posloupnost se opakuje od kroku dva. K resetu čítače dochází opět až při přechodu z kroku step1 (step_stop) do stavu step2. Výpis programů viz příloha na CD.

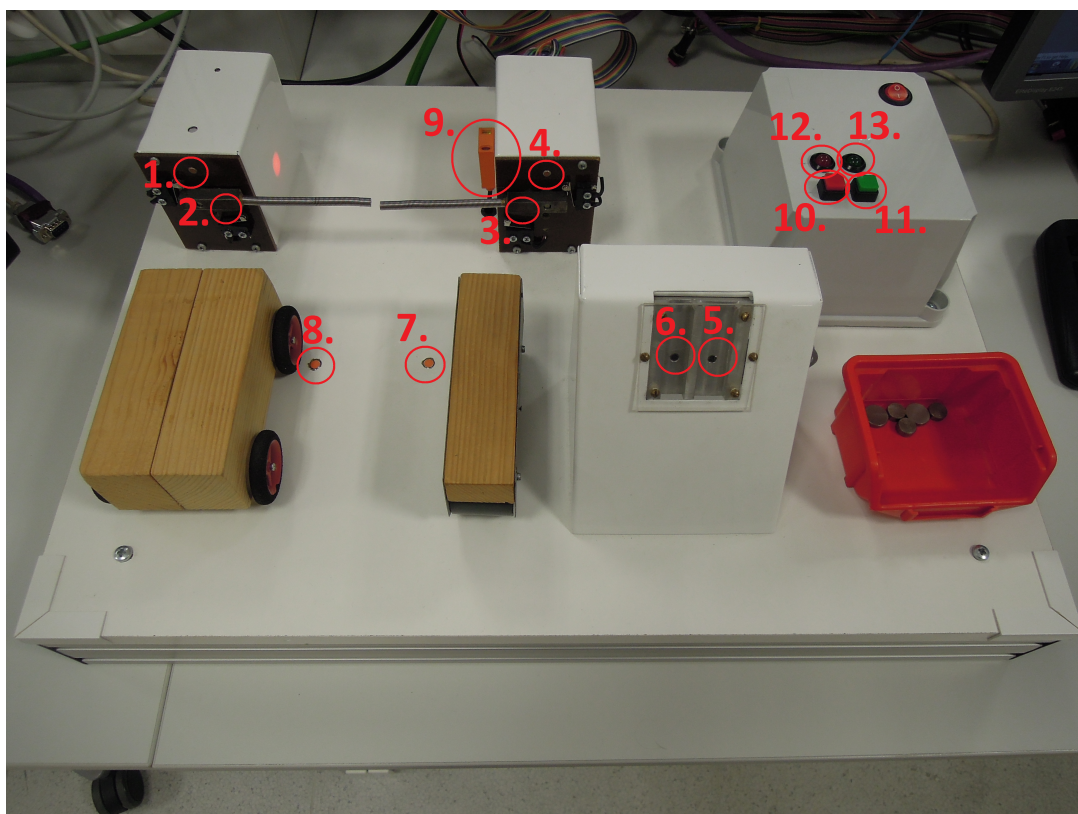
3.2.6 Závěr úlohy

Vzorové zadání je značně zjednodušené tím, že se uvažuje, že bedny přijíždějí vždy jen jedna po druhé s dostatečným odstupem. Kdyby se určilo, že jich může dojet více za sebou v kratším časovém intervalu, řešení by se velkou měrou zkomplikovalo. Ovšem pro takové zadání by musel být model jinak technicky vybaven. S nynějším stavem modelu je obtížné vymyslet složitější aplikace, jelikož model má určité nedostatky. Pro budoucí využití by bylo nejspíše vhodné model upravit. Na většině komponent je nějaký menší či větší nedostatek. Pásy nejsou dostatečně utažené tzn. mají moc velkou vůli a pozice beden je pak velmi nestabilní. Stabilitě nepřidává ani rychlost motorů. Motory nelze nijak regulovat, jejich chod je příliš hlučný a rychlý, což spolu s vůlí pásů činí velmi těžkým už jen pouhé položení beden na pás. Solenoidy mají krátký dosah. Kvůli tomu je nutné, aby byly umístěné tak, že zasahují do dráhy bednám, čímž se citelně zužuje prostor, na který je nutné bednu položit. Opět nutno dodat, že při rychlosti a vůli pásů to není jednoduché. Pokud se uživatel správně netrefí, je pak velká možnost, že se bedna zachytí o vyčnívající solenoid. V poslední řadě jsou tu pak senzory, které fungují uspokojivě. Levý senzor však někdy vynechává. Nejspíše kvůli houpání pásu nebo jeho poruše či špatné odrazové ploše. Model by bylo tedy nejspíše vhodné upravit. Při úpravách bych taktéž navrhl přidání několika čidel a závor pro zadržování/dávkování beden. Zadání by pak bylo možné vymyslet důmyslnější. Některé z těchto nedostatků jsem opravil. Provedenými úpravami se zabývám v kapitole 3.6.

3.3 Úloha č.2 - Parkoviště

3.3.1 Popis modelu

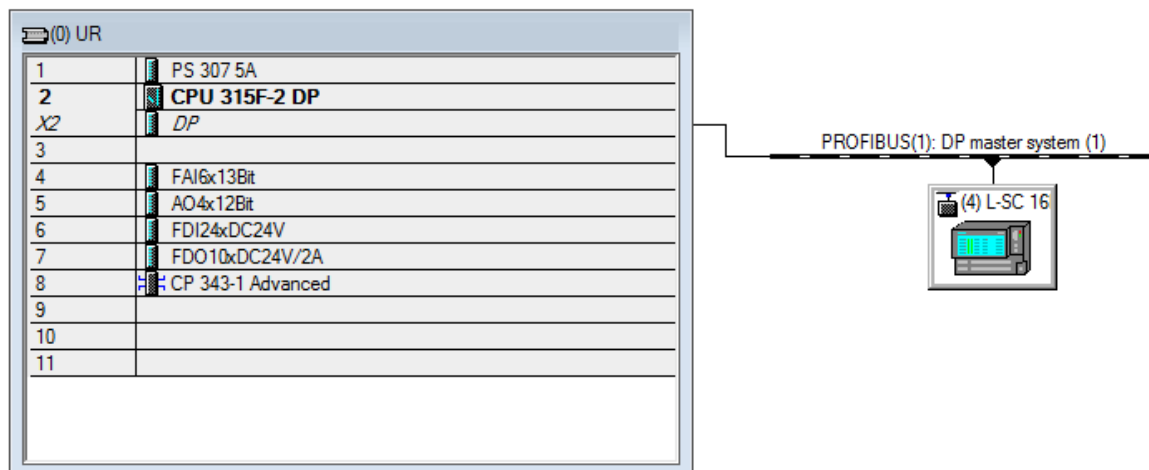
Model je zachycen na obr 3.11. Je tvořen bránou s dvěma závorami. Závory jsou ovládány dvěma motory. U obou závor jsou čtyři čidla, dvě hardwarové, které slouží jako vypínač motoru v krajní pozici a dvě programovatelné. Na jednom sloupku brány je připevněné optické čidlo. Napravo od brány je umístěný panel s dvěma tlačítky, dvěma LED diodami a hlavním vypínačem. Vlevo pod panelem je pak stojan s dvěma čidly sloužící jako pokladna. Spolu s modelem je pak třeba mít dva modely dopravních prostředků. Jeden větší pro demonstraci auta a druhé menší pro demonstraci motorky. Nutností jsou pak i oválné mince z kovu o dvou průměrech. Z fotky není patrné, kde se nachází konektor pro vstupy a výstupy modelu. Ten je umístěný za sloupkem s optickým čidlem. Popis pozic zaznačených na obrázku 3.11 je v tabulce 3.2. Tak jako předchozí model, i zde je použita univerzální řídicí deska viz obrázek 3.3 pro připojení elektrických prvků. Na řídicí desku jsou na příslušné vstupy připojeny i krajní čidla pro hardwarové zastavení motorů. Detailní schéma řídicí desky viz elektronická příloha na CD.



Obr. 3.11: Model parkoviště

3.3.2 HW konfigurace

Hlavní výpočetní jednotkou (Master) bylo PLC SIMATIC S7 300. Jako jednotku slave jsem zvolil modul ET200L-SC s následující konfigurací:



Obr. 3.12: Konfigurace PLC - lišta

Na obrázku 3.12 můžeme vidět, že konfigurace je shodná s přechozí úlohou tj. na první pozici na liště byl zdroj PS 307 5A, na druhé pozici je procesor 315F-2 DP. Třetí pozice je prázdná. Na čtvrté pozici je modul s analogovými vstupy FAI6x13bit, na páté pozici analogové výstupy AO4x12Bit. Následují moduly pro digitální vstupy a výstupy FDI24xDC24V a FDO10xDC24V/2A. Na poslední pozici je síťový modul CP 343-1 Advanced.

| Slot | Module | ... | Order number | Firm... | M... | I ... | Q... | Comment |
|------|-------------------|-----|---------------------|---------|------|--------|--------|---------|
| 1 | PS 307 5A | | 6ES7 307-1EA00-0AA0 | | | | | |
| 2 | CPU 315F-2 DP | | 6ES7 315-6FF01-0AB0 | V2.0 | 2 | | | |
| X2 | DP | | | | | 204... | | |
| 3 | | | | | | | | |
| 4 | FAI6x13Bit | | 6ES7 336-1HE00-0AB0 | | | 8...23 | 8...11 | |
| 5 | AO4x12Bit | | 6ES7 332-5HD01-0AB0 | | | | 272... | |
| 6 | FDI24xDC24V | | 6ES7 326-1BK00-0AB0 | | | 24...3 | 24...2 | |
| 7 | FDO10xDC24V/2A | | 6ES7 326-2BF01-0AB0 | | | 40...4 | 40...4 | |
| 8 | CP 343-1 Advanced | | 6GK7 343-1GX21-0XE0 | V1.0 | 3 | 448... | 448... | |
| 9 | | | | | | | | |
| 10 | | | | | | | | |
| 11 | | | | | | | | |

Obr. 3.13: Konfigurace PLC - výrobní čísla, adresy

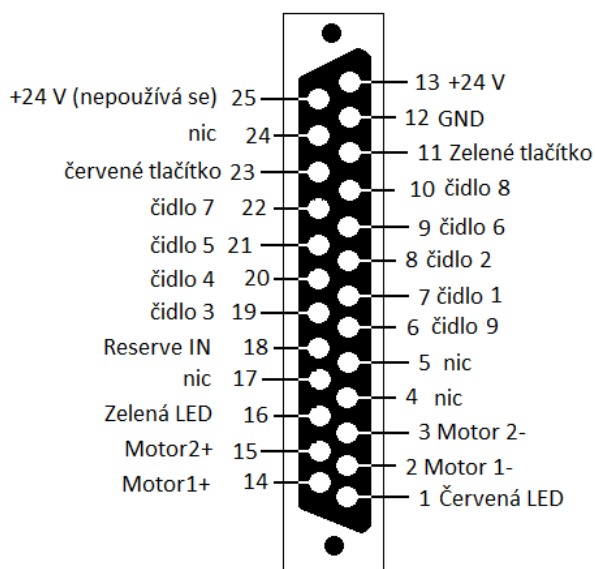
Pro mou aplikaci jsem nepoužil žádné z dostupných přídatných modulů pro analogové či digitální vstupy/výstupy. Použil jsem však digitální vstupy a výstupy decentralizované periferie, jejíž vstupní adresy začínají na nultém bytu, výstupní pak na druhém bytu, viz. obrázek 3.14.

| Slot | DP ID | Order Number / Designation | I Address | Q Address | Comment |
|------|-------|----------------------------|-----------|-----------|---------|
| 1 | 4 | Universal module | | | |
| 2 | 4 | Universal module | | | |
| 3 | 4 | Universal module | | | |
| 4 | 194 | Universal module | 0..1 | 2..3 | |
| 5 | 194 | Universal module | 6..7 | 6..7 | |

Obr. 3.14: Konfigurace DP - adresy

3.3.3 Zapojení výstupního konektoru modelu

Nutností pro splnění zadání bylo vytvořit protikus, kterým jsem pak následně spojil model s ET 200L-SC.



Obr. 3.15: Vývod z modelu - konektor CANON 25, zásuvka pohled shora

Propojení s DP je zaznačeno v tabulce 3.2. Jeden vodič plochého kabelu propojuje vždy příslušný pin s příslušnou svorkou, které pak odpovídá adresa a tag v programu uvedené v tabulce.

| Název | Svorka ET | Adresa | Tag v programu | Pozice viz. obr. 3.11 |
|------------------|-----------|--------|-------------------|-----------------------|
| Čidlo 1 | 17 | I0.0 | gate_left_top | 1. |
| Čidlo 2 | 18 | I0.1 | gate_left_bottom | 2. |
| Čidlo 3 | 19 | I0.2 | gate_right_bottom | 3. |
| Čidlo 4 | 20 | I0.3 | gate_right_top | 4. |
| Čidlo 5 | 21 | I0.4 | coin_large | 5. |
| Čidlo 6 | 22 | I0.5 | coin_small | 6. |
| Čidlo 7 | 23 | I0.6 | enter_right | 7. |
| Čidlo 8 | 24 | I0.7 | enter_left | 8. |
| Čidlo 9 | 25 | I1.0 | Entery | 9. |
| Červené tlačítko | 26 | I1.1 | btn_close | 10. |
| Zelené tlačítko | 27 | I1.2 | btn_open | 11. |
| Reserve in | 28 | - | - | - |
| Červená LED | 1 | Q2.0 | led_full | 12 |
| Zelená LED | 2 | Q2.1 | led_open | 13 |
| Motor 1+ | 3 | Q2.2 | mot_left_down | mezi 1. a 2. |
| Motor 1- | 4 | Q2.3 | mot_left_up | mezi 1. a 2. |
| Motor 2+ | 5 | Q2.4 | mot_right_up | mezi 3. a 4. |
| Motor 2- | 6 | Q2.5 | mot_right_down | mezi 3. a 4. |

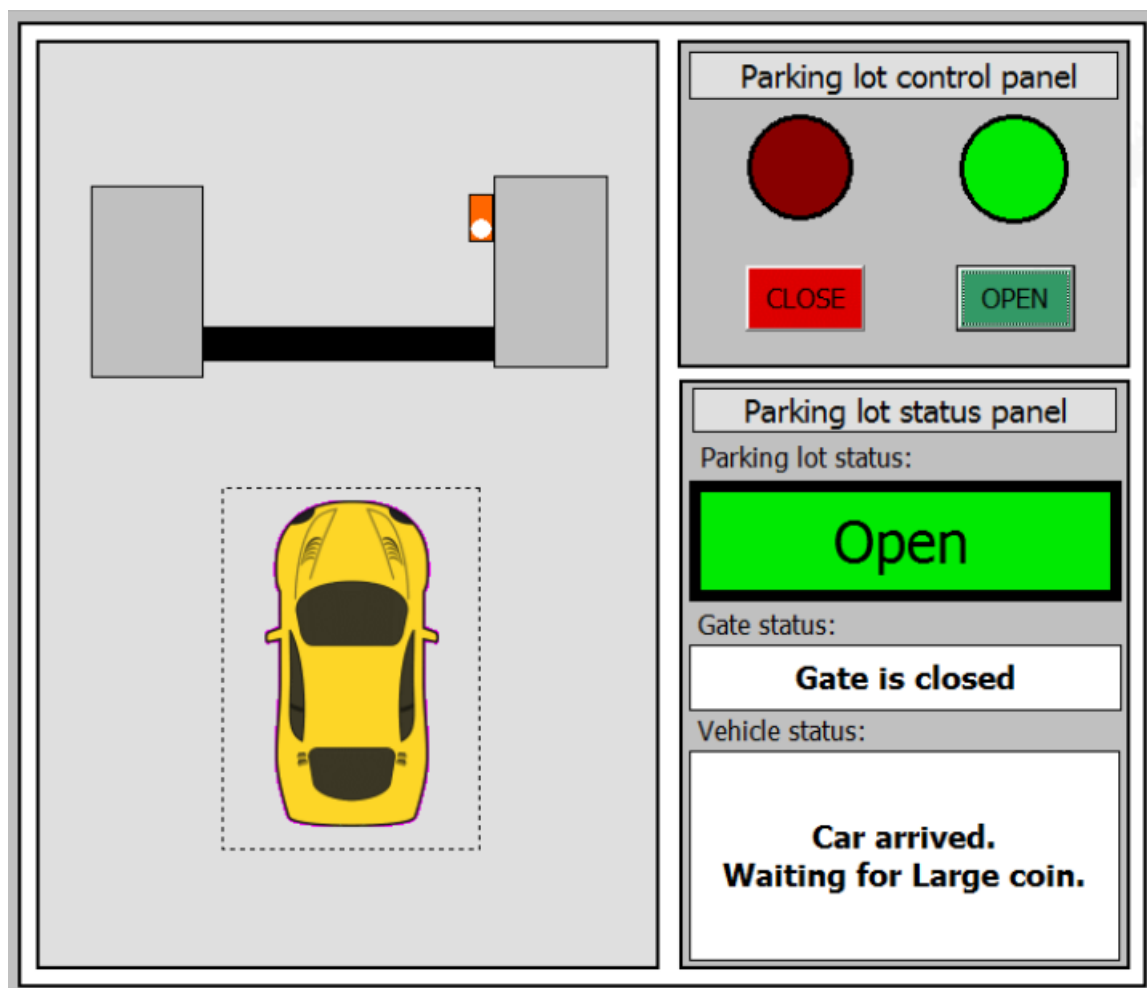
Tab. 3.2: Tabulka vstupů a výstupů a připojení na svorkovnice ET

3.3.4 Vizualizace

Vizualizace opět běží v runtime režimu na PC a byla vytvořena v programu WinCC flexible. Rozhraní je na obrázku 3.16.

Popis rozhraní vizualizace

Rozhraní bylo vytvořeno dle reálného modelu. V levé části je zachycen pohled shora na parkoviště. Dva sloupy se závorami mají vlastní animaci otvírání. Ta byla vytvořena pomocí dvou zrcadlených ukazatelů veličin zbaveného okrajů měrky a zbarvených tak, aby imitovaly závoru. Princip animace je uveden v následující kapitole. Na pravém sloupu je model optické závory, který imituje stav reálného čidla. Před závorami je vyznačen prostor pro příjezd vozidla, ve kterém se mění ikona vozidla podle toho, jaké vozidlo je přítomno na reálném modelu. Pokud před branou žádné není, objeví se zde nápis "No vehicle present". V pravé části je ovládací panel s informačními diodami a tlačítky, simulující reálný panel na modelu. V této části je i informační panel, ve kterém jsou uvedené informace o tom, zda je parkoviště otevřeno, v jakém stavu se nachází brána a taktéž informace o přítomném vozidle.



Obr. 3.16: Rozhraní vizualizace pro model parkoviště

Popis funkčního bloku pro ovládání vizualizace

Ovládání proměnných realizují ve funkčním bloku FB3. Je zde vyhodnocování stavu brány a vozidla. Každému stavu odpovídá číslo proměnné, dle které se pak zobrazuje daný prvek textového pole ve vizualizaci. Naprogramován je zde i mechanismus na ovládání animace otvírání brány. Ten funguje tak, že každých 100 ms se inkrementuje nebo dekrementuje čítač na základě toho, jestli se brána otvírá či zavírá. Tato hodnota je následně zobrazena na ukazatelích veličin v podobě sloupců obrácených zrcadlově proti sobě do horizontální polohy. Viz soubor animace modelu v elektronické příloze na CD.

3.3.5 Vzorová úloha a řešení

Zadání vzorové úlohy

Navrhněte řízení modelu parkoviště. Parkoviště skýtá bránu, která se po příjezdu vozidla a po vhození odpovídající mince do pokladny otevře. Celé řízení navrhněte s použitím decentralizované periferie ET 200L-SC. Musí platit následující:

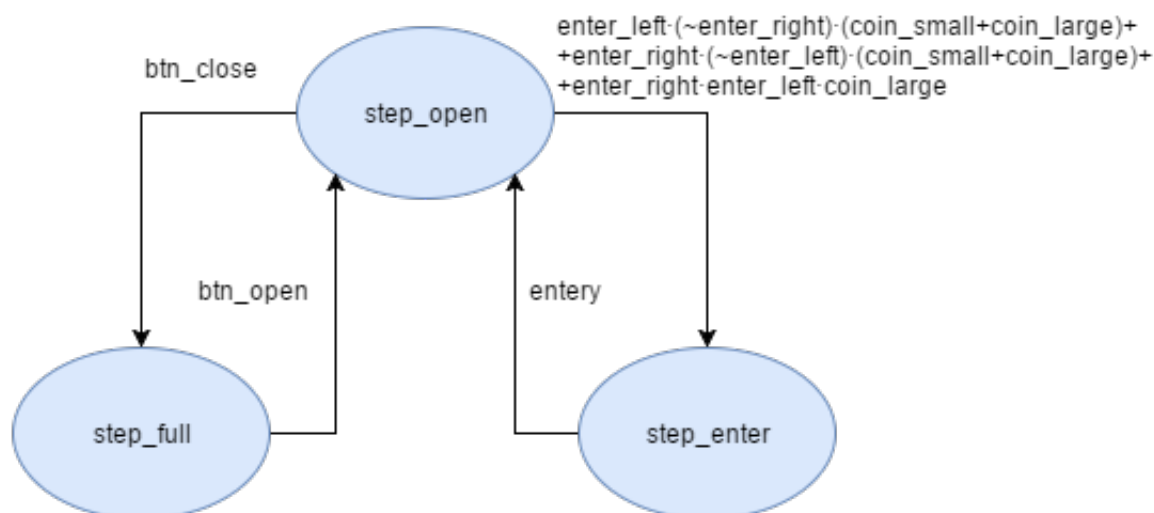
- Při příjezdu motorky (malý model) se otevře brána při vhození malé nebo velké mince.
- Při příjezdu auta (velký model) se otevře brána pouze při vhození velké mince.
- Brána se nesmí zavřít dříve než vozidlo opustí vjezd. Neplatí to však pokud vozidlo vycouvá pryč. V takovém případě zůstane brána otevřená.
- Přijímání vozidel je možné jen v době pokud je na parkovišti volno. To lze ovládat pomocí červeného a zeleného tlačítka.
- Pokud je volno, svítí zelená kontrolka. Pokud je plno, svítí červená kontrolka.
- Navrhněte otvírání brány tak, aby se závory zastavovaly ve správných pozicích tj. na programovatelných čidlech. Vykompenzujte nepřesnost spínání indukčních snímačů při zavírání závor tak, aby se obě ramena závor zastavily vodorovně a naproti sobě.

Popis bloků v programu

Program obsahuje 6 funkčních bloků a jeden organizační, ve kterém jsou potřebné funkční bloky volány. V bloku FB1 je naprogramováno v jazyku LAD ovládání motorů a kompenzace zastavení spínání indukčních čidel viz. obrázek 3.18. Je zde i řešení zavírání brány až po průjezdu auta. Ve funkčním bloku 2 je v jazyce LAD implementován stavový automat dle stavového diagramu viz. 3.17. Ve funkčním bloku 3 je implementováno v jazyce LAD řízení proměnných potřebných pro ovládání vizualizace. Tento blok je společný pro zbylé jazyky a je tedy naprogramován pouze v jazyce LAD. FB5 a FB6 jsou obdobou FB1 a FB2 s tím rozdílem, že jsou naprogramovány v jazyku STL. Přístup k řešení problému nebyl pro tyto jazyky rozdílný. Rozdílný přístup si vyžadoval blok FB4 naprogramovaný v jazyku GRAPH viz. kapitola "Popis stavového automatu v jazyku GRAPH". Kompletní výpis programů je uveden v elektronické příloze na CD.

Popis stavového automatu v jazycích LAD a STL

Popis je pro oba jazyky obdobný. Na začátku se program nachází buď ve stavu `step_open` nebo ve stavu `step_full`. Mezi těmito stavy se může přepínat pomocí červeného nebo zeleného tlačítka. Zmáčknutím červeného tlačítka může uživatel prohlásit, že je parkoviště plné, čímž se program dostane do stavu `step_full` a rozsvítí se

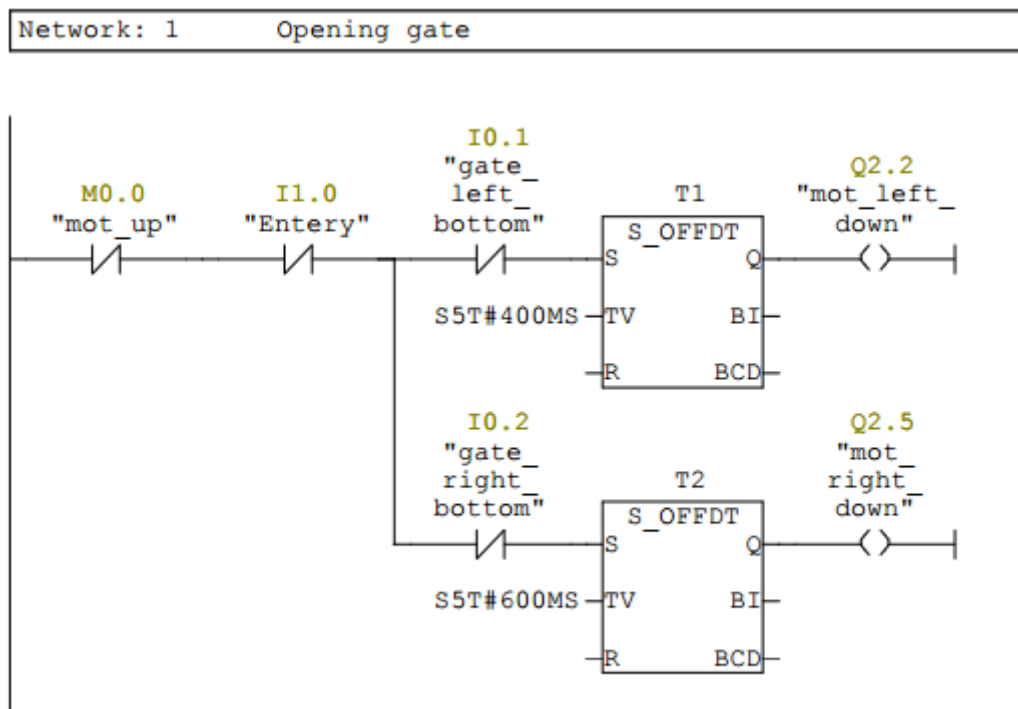


Obr. 3.17: Stavový diagram pro řízení modelu parkoviště

červená dioda na znamení toho, že je parkoviště plné. Zároveň v tomto stavu nelze otevřít bránu. Po zmáčknutí zeleného tlačítka se parkoviště otevře, tzn. program se dostane do stavu `step_open`, ve kterém svítí zelená dioda a brána čeká na příjezd vozidla. Snímači v podlaze (čidlo 7 a 8) program pozná, zdali došla motorka či auto. Po příjezdu program čeká na vhození vhodné mince. Po vhození se program dostane do stavu `step_enter`, ve kterém se otevře závora a program čeká, až se sepně čidlo 9. Po sepnutí program vyhodnotí, zda auto už projíždí bránou a dostane se zpět do stavu `step_open`. Závora se ovšem ještě nezavírá. V bloku FB1 nebo FB5 je podmínka (viz. obr. 3.18), že se závora začne zavírat až v době, kdy čidlo 9 není sepnuté, tzn. auto odjelo z dosahu závory.

Popis stavového automatu v jazyku GRAPH

Z důvodu rozdílů v principech jazyka od jazyku STL a LAD, bylo potřeba zvolit jiný přístup. Neboť v jazyku GRAPH už nebylo možné naprogramovat ovládání motorů brány tak, jak jsem učinil v jazycích LAD a STL. Celý problém jsem si musel rozdělit na 12 kroků. V prvním kroku se otvírá parkoviště. Z tohoto kroku se můžeme při splnění příslušných podmínek dostat buď do kroku, kde se parkoviště zavře nebo do série kroků obsluhujících vjezd vozidla. Při vykonávání téhle posloupnosti kroků nelze parkoviště zavřít. Po skončení posloupnosti program přejde opět do prvního kroku. Z kroku, kde je parkoviště zavřené, se lze dostat do prvního kroku pouze otevřením parkoviště (stlačení zeleného tlačítka buď na modelu nebo ve vizualizaci).



Obr. 3.18: Kompenzace a řízení motoru závor.

3.3.6 Závěr úlohy

V úloze si student procvičí používání stavového automatu a časovačů. Úloha z pohledu hardwaru nepotřebuje žádné vylepšení. Co se týče samotného zadání, tak by šlo různě obměnit či vylepšit. Jako třeba podmínkou, že při vycouvání vozidla z vjezdu se brána sama zavře po nějaké době, kdy auto už nebude ve vjezdu. Nebo třeba počítáním vjetých aut, přičemž by program sám poznal, kdy je parkoviště plné a kdy má ještě místo. V tomto případě by bylo nutné vymyslet způsob výjezdů aut. Buďto programově tím samým vjezdem, což by bylo složité, nebo jen nasimulovat v podobě nějakého tlačítka (stisk = jedno vozidlo vyjelo). Kompletní výpisy programu viz. elektronická příloha na CD.

3.4 Porovnání programovacích jazyků

Jako základní jazyk byl zvolen jazyk LAD pro jeho názornost a jednoduchost. Nutno dodat že názornost jazyku LAD je omezená rozsahem práce. Při rozsáhlých projektech může jazyk LAD být hůře čitelný než například STL. Ovšem mé zadání není rozsáhlé, proto byl jazyk LAD použit i na funkční bloky pro řízení vizualizace. A to především pro jednodušší práci při porovnávání, konverzi nebo jiné manipulaci s proměnnými, kdy pomocí celých bloků je jejich implementace do kódů jednodušší.

V případě použití STL lze provádět stejné akce, ale naučení způsobu provedení těchto operací je zdoluhavější. Jinými slovy, programování v jazyku LAD je více intuitivní. Ovšem pokud se programátor naučí jazyk STL řádně používat, tak je pak programování příjemnější z důvodu a to, že programátor používá pouze klávesnici a nemusí sahát na myš. Programování je pak efektivnější a rychlejší. A jak již bylo zmíněno, rozsáhlý program je ve výsledku přehlednější a čitelnější, než kdyby byl vytvořen v jazyku LAD.

Je vhodné také zmínit, že původní záměr s jazyky STL a LAD byl používat je pro programy na základě pravdivostních tabulek a logických obvodů či reléových schémát. To tedy znamená, že nebyly vytvořeny speciálně pro stavové diagramy tak, jako jazyk GRAPH. Ovšem vytvoření stavového diagramu v jazyku STL nebo LAD je vcelku jednoduchou záležitostí. Výhoda LAD a STL oproti jazyku GRAPH je v jejich paralelnosti, kdy například u modelu parkoviště byla tato vlastnost využita pro řízení otevírání brány. Proto jazyk GRAPH není úplně vhodný v případě, kdy je potřeba některé proměnné neustále sledovat.

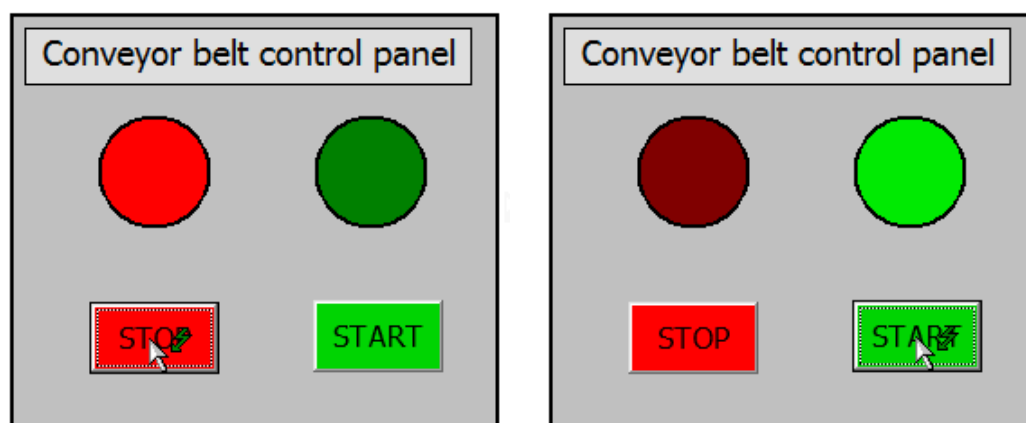
Ovšem u modelu dopravního pásu se jazyk GRAPH jevil jako nejvhodnější. Zejména pro jeho názornost a jednoduchost. Je sice nutné přistupovat k problematice trochu jinak, ale pokud lze program řešit stavovým automatem, pak je jazyk GRAPH mnohdy vhodnější než LAD či STL. Ovšem pro funkční blok na řízení vizualizace se jazyk GRAPH nehodí takřka vůbec. Řízení vizualizace se totiž dá rozdělit do jednotlivých kroků jen velmi těžko, né-li téměř vůbec, a to díky mnoha operacím, které musejí probíhat paralelně.

Ve výsledku je pak závěr porovnání takový, že jako nejlepší volba je tyto jazyky kombinovat. Oba modely lze poměrně jednoduše naprogramovat ve všech třech jazycích. Jen u modelu parkoviště bylo řešení pomocí jazyku GRAPH krkolomnější. Použití jazyku LAD či STL je zde vhodnější. Model dopravníku beden je pak dobrý pro demonstraci jazyku GRAPH, ale s jeho použitím je řešení možná až moc jednoduché.

3.5 Demonstrace funkčnosti vizualizace

3.5.1 Model dopravníku beden

Po připojení úlohy k DP ET200B, nahrání programu do PLC SIMATIC S7 300 a po spuštění vizualizace na PC v programu WinCC, fungovala vizualizace následovně. Jako první můžeme vidět na obrázku 3.19 ovládací panel. Důsledek zmáčknutí příslušného tlačítka je, že se rozsvítí nebo zhasnou příslušné informační LED diody dle programu nahraného v PLC. Po stisku červeného tlačítka se taktéž zastaví pásy.

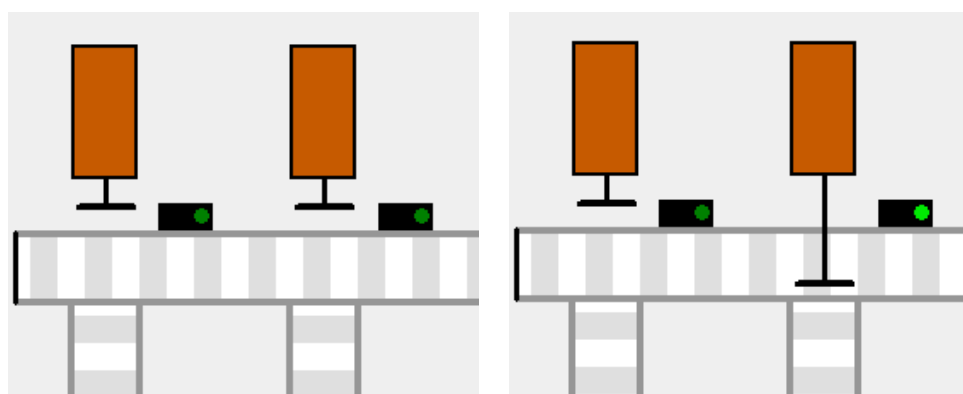


(a) Vypnutý stav.

(b) Zapnutý stav.

Obr. 3.19: Kontrolní panel z pravého horního rohu rozhraní.

Stisknutím zeleného tlačítka se resetuje stav čítače beden a pásy se dají do pohybu. Na obrázku 3.20 vidíme část se solenoidy a senzory. Senzory se rozsvítí v přítomnosti bedny ve snímané oblasti na reálném modelu. Následně se po určitém čase sepne so-

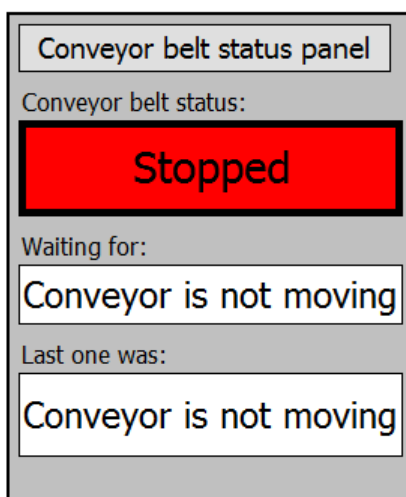


(a) Neseprnutý stav.

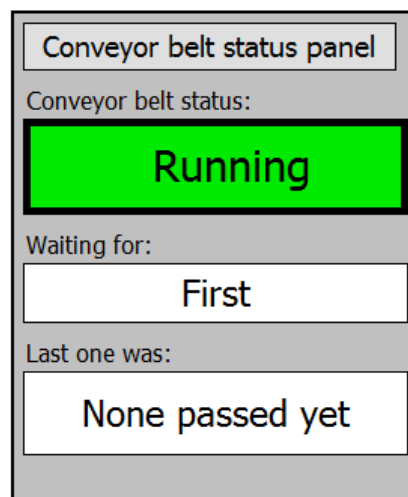
(b) Sepnutý stav.

Obr. 3.20: Ukázka sepnutí solenoidu a senzoru.

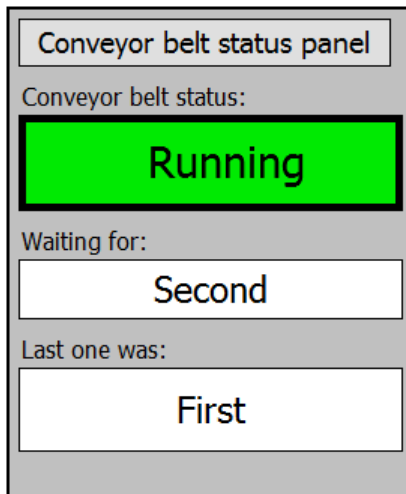
lenoid. Na obrázku 3.20(b) je zachycen moment, kdy je sepnuté čidlo i solenoid. Na skupině obrázků 3.21 jsou uvedeny příklady textů zobrazovaných na panelu stavů dopravního pásu. Na 3.21(a) je zobrazena situace, kdy je pás zastavený. Na 3.21(b) pak když je právě spuštěný a ještě nebyla detekována žádná bedna. Na 3.21(c) je situace po detekování první bedny. Na 3.21(d) je situace po detekování třetí bedny. Kompletní animace chování modelu je v elektronické příloze na CD.



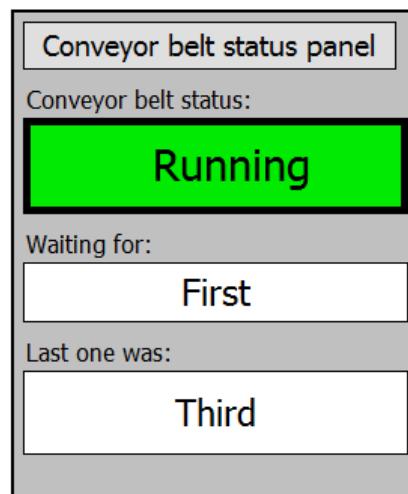
(a) Vypnutý stav.



(b) Stav krátce po zapnutí pásů.



(c) Detekce první bedny

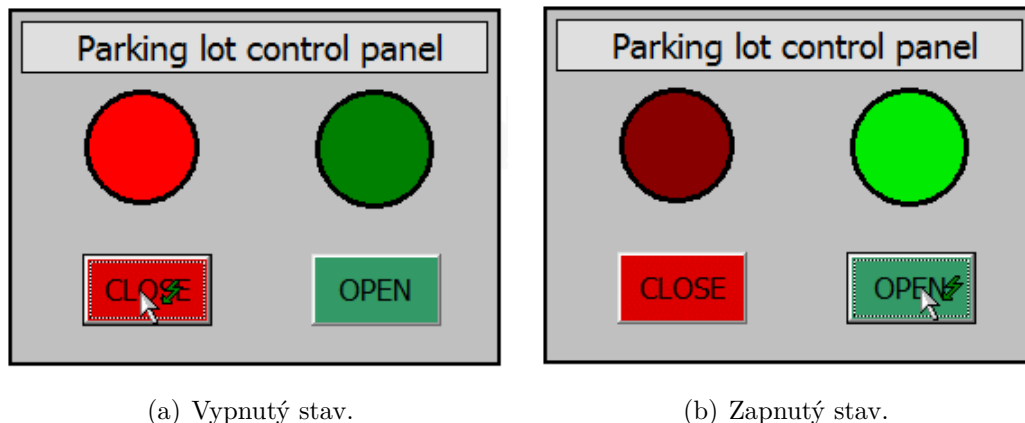


(d) Detekce třetí bedny

Obr. 3.21: Kontrolní panel z pravého horního rohu rozhraní.

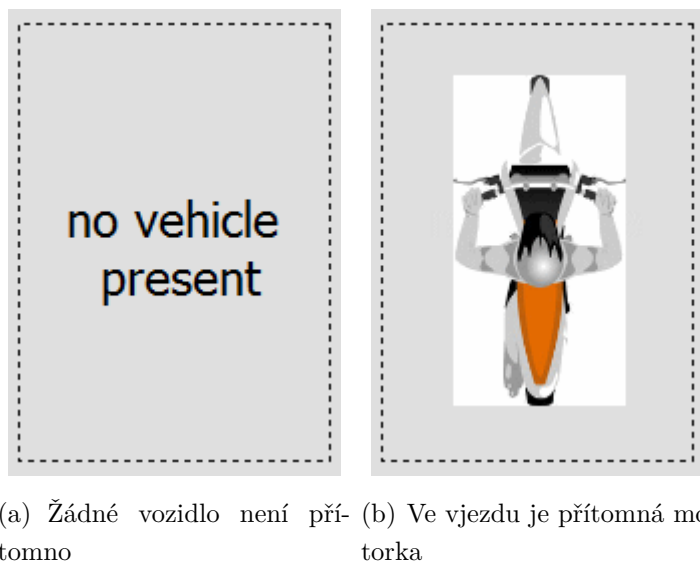
3.5.2 Model parkoviště

Po připojení úlohy k DP ET200L-SC, nahrání programu do PLC SIMATIC S7 300 a po spuštění vizualizace na PC v programu WinCC, fungovala vizualizace následovně. Jako první můžeme vidět na obrázku 3.22 ovládací panel. Důsledek zmáčknutí příslušného tlačítka je, že se rozsvítí nebo zhasnou příslušné informační LED diody dle programu nahraného v PLC. Tlačítka ovlivňují i samotný model.



Obr. 3.22: Kontrolní panel z pravého horního rohu rozhraní.

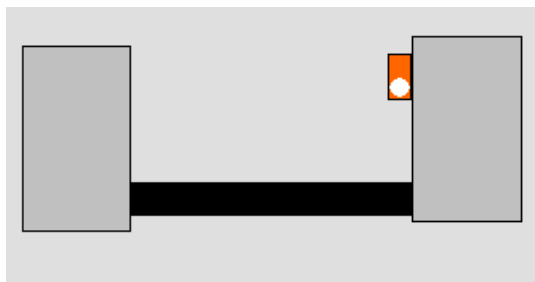
Na obrázcích 3.23 je ukázáno, jak vypadá reprezentace nepřítomnosti vozidla či přítomnosti motorky. Jak vypadá přítomnost auta je znázorněno na obrázku 3.25(a).



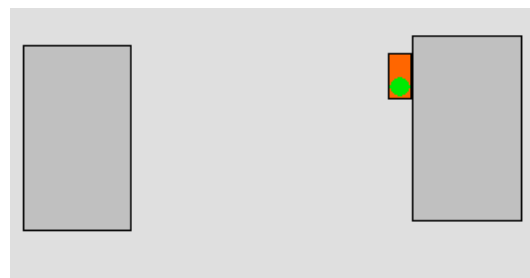
Obr. 3.23: Reprezentace vozidel před bránou.

Na obrázcích 3.24 je vyobrazeno jak vypadá brána, když jsou závory uzavřené nebo otevřené. Na obrázku 3.25(a) je ukázána situace, kdy se brána právě otevírá.

Na obrázku 3.25(b) je zaznamenán průjezd vozidla otevřenou bránou.

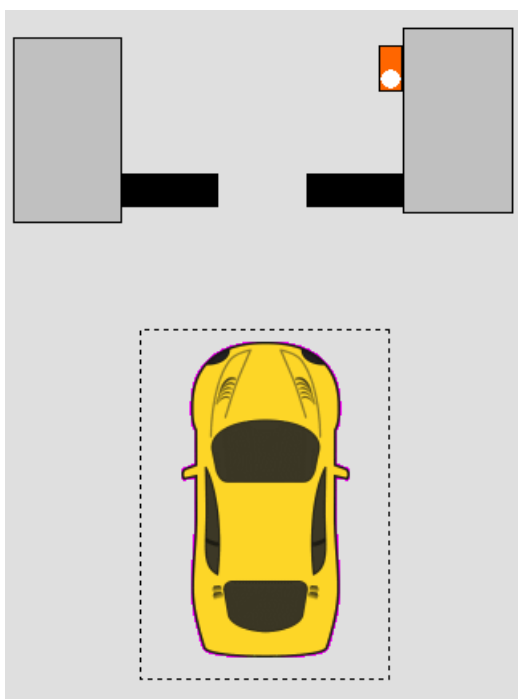


(a) Zavřené závory.

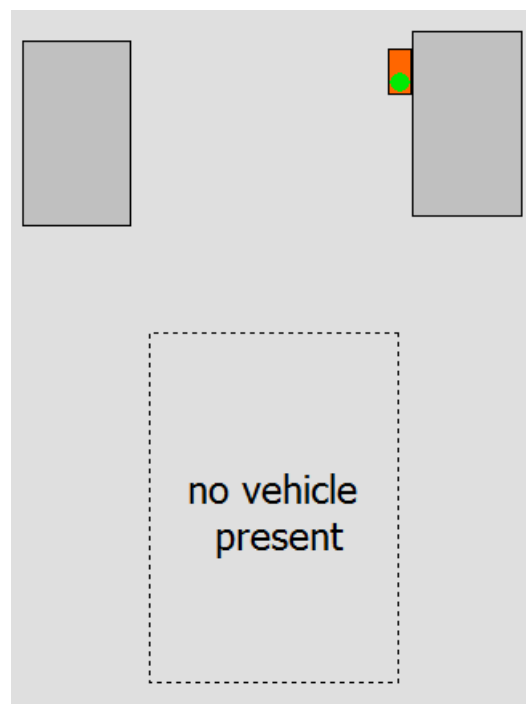


(b) Otevřené závory a indikace průjezdu auta.

Obr. 3.24: Vyobrazení stavu závor.



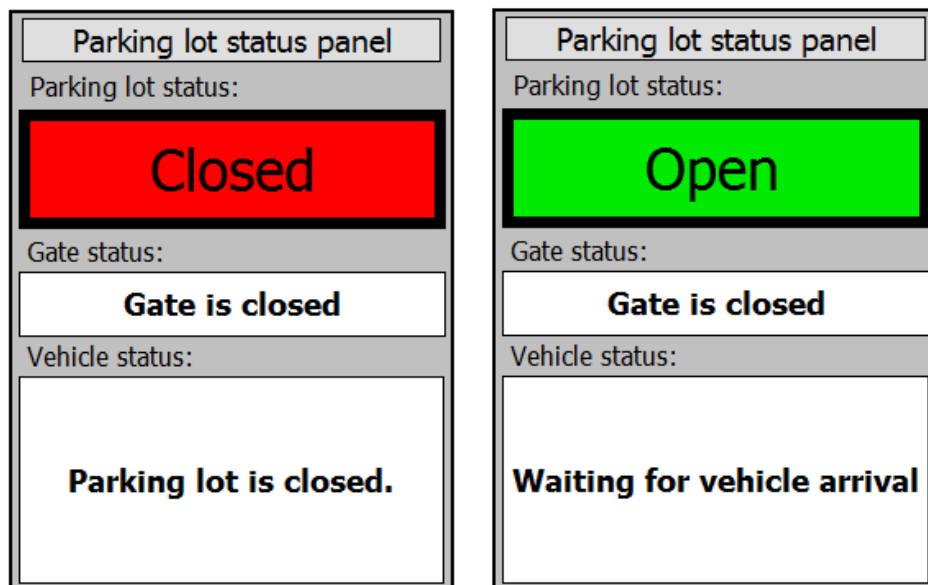
(a) Vozidlo čekající na otevření brány.



(b) Průjezd vozidla

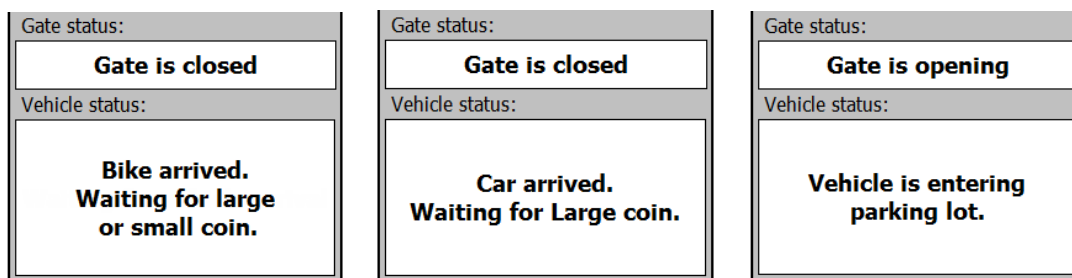
Obr. 3.25: Průjezd vozidla.

Na obrázcích 3.26 je zachycen kontrolní panel ve stavu zavřeného parkoviště a ve stavu, kdy je parkoviště otevřené. Na skupině obrázků jsou zaznamenány textové hlášky o stavu vozidla a brány, při kterých je parkoviště otevřeno. Kompletní animace chování modelu je v elektronické příloze na CD.



(a) Parkoviště je uzavřeno.

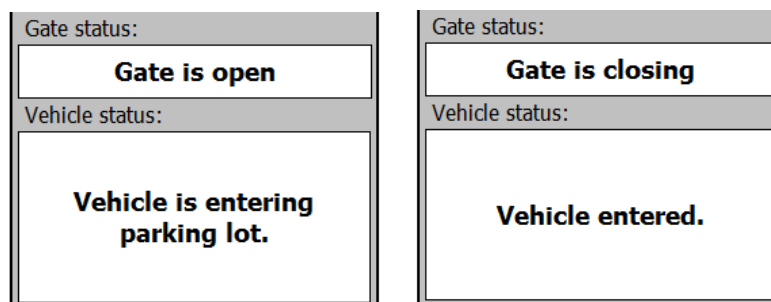
(b) Parkoviště je otevřeno, brána je zavřená, čeká se na příjezd vozidla.



(c) Přítomnost motorky.

(d) Přítomnost auta

(e) Po vhození mince.



(f) Brána je otevřená.

(g) Vozidlo vjelo na parkoviště.

Obr. 3.26: Textové hlášky o stavu parkoviště, brány a vozidla

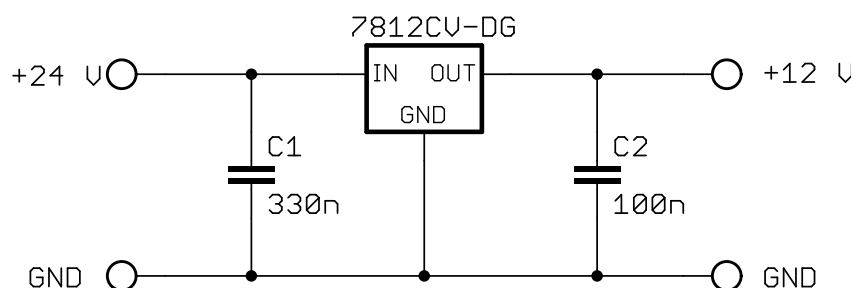
3.6 Úpravy modelu dopravníku beden

Na základě závěru úlohy dopravníku beden bylo uznáno za vhodné opravit některé z neduhů v něm zmíněných. Důkladnou analýzou těchto neduhů modelu se zjistilo, že příčina hluku je zejména přenášení vibrací z motorů na stěny dopravníku a následně pak do dřevěné podložky posazené na rámu zkonstruované z hliníkových profilů, která funguje jako ozvučná skříňka zesilující vibrace. Bylo tedy potřeba snížit generované vibrace a odizolovat přenos vibrací do podložky.

Tento efekt byl částečně potlačen vložením podložek z plstěné látky mezi motory a stěnu dopravníků, a také do kontaktních ploch mezi stěnou dopravníků a dřevěnou základnou. Vždy u každého šroubu dvě podložky. Jedna z horní strany a druhá ze strany matice. Tím se částečně ztlumil přenos vibrací v místech šroubů. Díky těmto podložkám se celá stěna dopravníku zvedla, čímž vznikla mezera mezi stěnou a dřevěnou základnou, což mělo za následek, že se v těchto místech vibrace nepřenášely téměř vůbec.

Dalším neduhem byla nestabilita pásů. Ta byla eliminována doplněním kovových válečků, které byly umístěny rovnoměrně pod pásy, díky čemuž se částečně pás napnul a zároveň ho podepřely. Válečky měly ještě jeden pozitivní důsledek, a sice že srovnaly stěny dopravníků vůči sobě a tudíž už nedocházelo k posunutí jedné stěny vůči druhé, což vedlo k tomu, že motor měl díky zešíkmené ose nepravdělnou zátěž. To byla příčina kolísavého zvuku a zesílení hluku, protože motor musel zvýšit svůj výkon a následně opět snížit.

K podepření pásu bylo použito i plexisklo v místech, kde z hlavního pásu vedou pásy vedlejší. Plexisklo je z jedné strany položeno na okrajích stěn vedlejšího dopravníku a z druhé strany přišroubováno ke stěně hlavního dopravníku. Jeden z problémů úlohy byla také mezera mezi pásem a stěnou. Tento nedostatek byl vyřešen přilepením zešíkmeného plexiskla na stěnu při tom bylo do plexiskla nutné vyfrézovat otvory pro válečky.



Obr. 3.27: Zapojení stabilizátoru pevného napětí STMicroelectronics 7812CV-DG

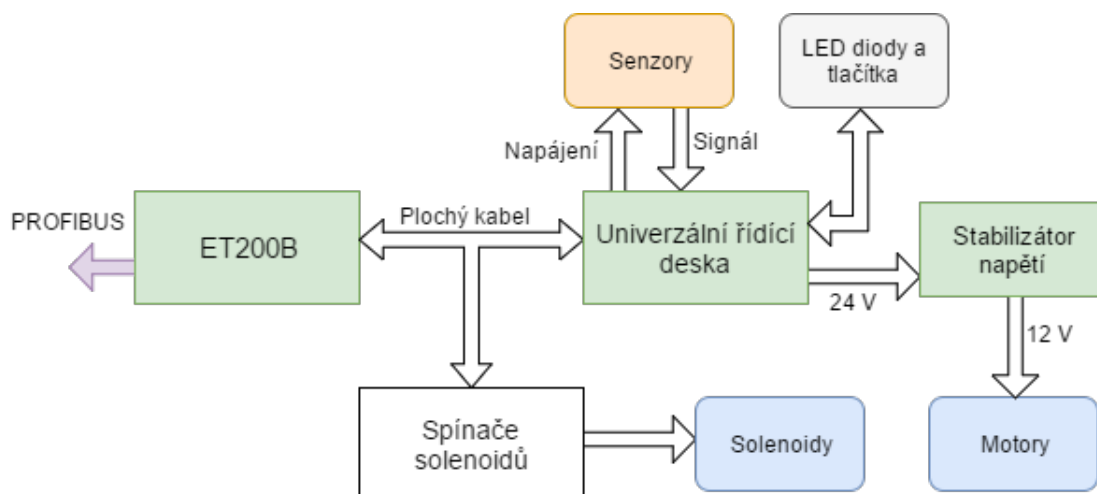
S nestabilitou beden na pásu a také částečně s hlukem souvisely použité motory

pohánějící pásy, resp. jejich rychlost. Vzhledem k tomu, že se jedná o klasické stejnosměrné motory, bylo snížení otáček motoru docíleno snížením napájecího napětí motorů pomocí napětového stabilizátoru, viz obr 3.27, který původních 24 V snížil na 12 V. Byly použity dva stabilizátory, jeden na hlavní motor, druhý na dva vedlejší.

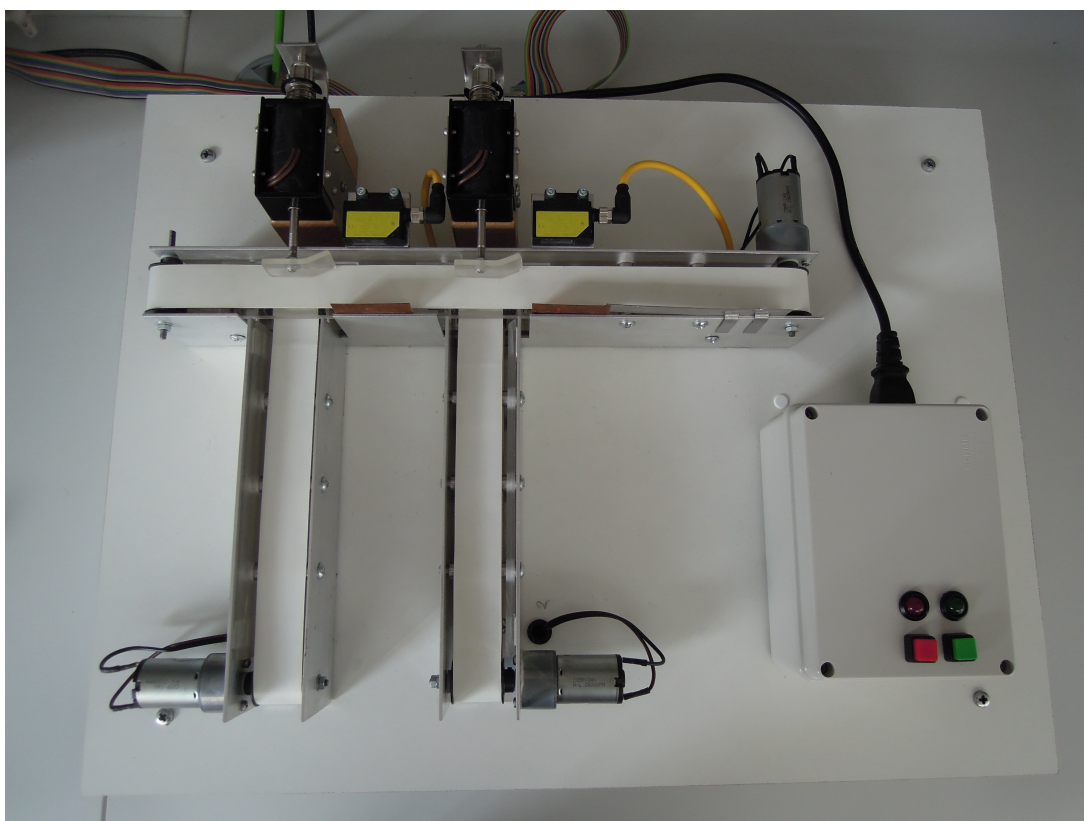
Jako poslední byl řešen problém s možností zastavení bedny o vyčnívající rameno solenoidu nebo zaklínění do mezery mezi pás a stěnu. Tento problém byl vyřešen tak, že se do mezery mezi pás a stěnu nalepilo zešíkmené plexisklo, které vymezilo bednám cestu. U solenoidu bylo řešení složitější. Kvůli krátkému dosahu solenoidů, zde nebyla možnost posunout solenoidy dál. Proto se muselo zvolit jiné řešení. To spočívalo v tom, že se zakončení ramene z plexiskla natvarovalo tak, aby v případě, kdy je bedna mimo dosah, došlo k tomu, že ohnutá ploška navede bednu před rovnou část plexiskla.

Poslední změnou, spíše estetickou, byla výměna průmyslové krabičky, která obsahuje napájecí zdroj, ovládací tlačítka a signalizační LED diody, protože původní krabice byl pouze obrácený spodní díl jiné krabičky, což nebudilo dobrý dojem, protože na spodní strany byly vidět stopy po lisování plastu. Výměna krabičky spočívala v koupi nové, do které byly vyřezány otvory pro diody, tlačítka a jednoho nového otvoru pro euro konektor k napájení. Z toho důvodu byl vyměněn i kabel pro napájení za standardní kabel k PC.

Z důvodů lepší manipulace s modelem byly provedeny i další menší změny, jako třeba organizace kabeláže nebo přidání silikonové podložky mezi krajní válce a stěnu držící pás pro lepší skluz. Díky tomu došlo k zlepšení chodu pásu a tudíž i snížení zátěže na motory. Při zreorganizování rozvodů modelu bylo zjištěno, že jedna z řídicích desek je zde jen pro řízení jednoho z motorů pohánějícího sekundární pásy, které vždy běží paralelně. Pro účely této úlohy je tak dostačující ovládat motory obou sekundárních pásů pouze jedním výstupem. Zapojení úlohy pomocí univerzální desky lze vidět na obrázku 3.28. Detailní schéma desky viz elektronická příloha na CD. Výsledná podoba modelu je zobrazena na obrázku 3.29



Obr. 3.28: Blokový diagram zapojení univerzální desky u modelu dopravníku beden.



Obr. 3.29: Výsledná podoba modelu dopravníku beden.

4 ZÁVĚR

Cílem mé práce bylo porozumět a popsat laboratorní modely, vytvořit možnost pro jejich rychlé připojení na decentralizovanou periferii, vymyšlení ukázkových zadání a jejich naprogramování, vytvoření vizualizace a zhodnocení modelů. Mimo tyto cíle jsem dosáhl i úpravy dopravníku beden, což nebylo součástí zadání, ale okolnosti vedly k tomu, že úprava byla nutná. Jak jsem již výše zmínil, výsledkem mé práce je možnost rychlého přepojení modelů mezi připojením vstupů/výstupů přímo na PLC anebo připojením na decentralizovanou periferii, a dále zjištění nedostatků modelů a zhodnocení jejich využitelnosti pro výuku. Model parkoviště je navržen a zkonstruován kvalitně. Ukázkové zadání je dle mého názoru vhodné pro výuku, jelikož v něm lze procvičit různé aspekty programování na PLC. Jako základní programovací jazyk pro výuku bych volil LAD nebo STL pro jejich paralelnost. Pro model dopravníku beden bych volil GRAPH pro názornou aplikaci stavového automatu. Dopravník beden byl upraven a v nynějším stavu je uzpůsobený pro náhradu některé z úloh. V modelu dopravníku beden si student může procvičit naprogramování stavového automatu, či zkusit použití čítačů a časovačů. Po úpravách je model dopravníku beden ve stavu použitelném pro výuku, byť je jeho řešení poměrně jednoduché. V obou úkolech si student procvičí vytvoření hardwarové konfigurace nejen samotného PLC, ale i decentralizované periferie. Oba modely byly naprogramovány ve třech jazycích a byla vytvořena vizualizace pro ovládání a monitorování úloh na PC.

LITERATURA

- [1] ZEŽULKA, F. *Prostředky průmyslové automatizace*. Brno, 2004
- [2] ZEŽULKA, František, Petr FIEDLER, Petr VAŇOUS a Petr Cach. *Průmyslové komunikační sítě*. Brno, 2000
- [3] SIEMENS AG. *ET 200L, ET 200L-SC and ET 200L-SC IM-SC Distributed I/O Device - Manual* [online]. EWA 4NEB 780 6009-02c, Edition 04. Federal Republic of Germany. [cit. 6. 1. 2017]. <https://cache.industry.siemens.com/dl/files/528/23988528/att_47365/v1/ET200L_e.pdf>.
- [4] SIEMENS AG. *ET 200B Distributed I/O Station - Manual* [online]. EWA 4NEB 812 6089-02c, Edition 04. Federal Republic of Germany. [cit. 6. 1. 2017]. Dostupné z URL: <https://cache.industry.siemens.com/dl/files/452/1144452/att_34041/v1/ET200B_e.pdf>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

| | |
|----------|---|
| DBFT | Deutsches Bundesministerium für Forschungs und Technik |
| DP | Decetralizovaná Periferie nebo taktéž Distributed Periphery |
| SC | Smart Connect |
| FB | Funkční blok |
| OB | Organizační blok |
| DI | Digital Input |
| DO | Digital Output |
| AI | Analog Input |
| AO | Analog Output |
| PA | Process Automation |
| PROFIBUS | Process field bus |
| LAD | Ladder Diagram |
| STL | Statement list |
| PLC | Programovatelný logický automat (Programmable Logic Controller) |
| CD | Compact Disc |
| PC | Personal Computer |
| V | Volt |
| Hz | Hertz |
| obr. | obrázek |
| např. | například |
| tj. | to je |
| č. | číslo |
| ms | milisekunda |
| tl. | tlačítko |
| HW | Hardware |

A OBSAH PŘÍLOŽENÉHO CD

```
/ ..... koreňový adresář přiloženého CD
├─ Bakalářská práce Manc1 2017.pdf
├─ Výpisy z programu modelu dopravníku beden
│   └─ Symbol table.pdf.
│       └─ OB1.pdf
│           └─ FB1_LAD.pdf
│               └─ FB2_VIZ.pdf
│                   └─ FB3_STL.pdf
│                       └─ FB4_GRAPH.pdf
├─ Výpisy z programu modelu parkoviště
│   └─ Symbol table.pdf
│       └─ OB1.pdf
│           └─ FB1_LAD_MOT.pdf
│               └─ FB2_LAD_STEPS.pdf
│                   └─ FB3_VIZ.pdf
│                       └─ FB4_GRAPH.pdf
│                           └─ FB5_STL_MOT.pdf
│                               └─ FB6_STL_STEPS.pdf
├─ Vizualizace
│   └─ Výpis z vizualizace modelu dopravníku beden.pdf
│       └─ Výpis tagů z vizualizace modelu dopravníku beden.pdf
│           └─ Výpis z vizualizace modelu parkoviště.pdf
│               └─ Výpis tagů vizualizace modelu parkoviště.pdf
│                   └─ Animace vizualizace modelu dopravníku beden.gif
│                       └─ Animace vizualizace modelu parkoviště.gif
├─ Program pro STEP7 modelu dopravníku beden..... Složka obsahuje celý
│   program i s vizualizací
├─ Program pro STEP7 modelu parkoviště. Složka obsahuje celý program i s
│   vizualizací
├─ Univerzální řídící deska
│   └─ Schéma desky.png
│       └─ obrázek desky.png
```